WDL-TR 2184

15 NOVEMBER 1963

temp
19 4740

PROGRAMMER'S MANUAL for Interplanetary Error Propagation Program

CONTRACT NAS 5-3342

prepared for

National Aeronautics and Space Administration
Goddard Space Flight Center
Greenbelt, Maryland

PHILCO | WDL DIVISION

A SUBSIDIARY OF Ford Motor Company | PALO ALTO, CALIFORNIA

7

RQ+-18040

WDL-TR2184
15 November 1963

PROGRAMMER'S MANUAL FOR INTERPLANETARY ERROR PROPAGATION PROGRAM

Prepared by

PHILCO CORPORATION
A Subsidiary of Ford Motor Company
WDL Division
Palo Alto, California

CONTRACT NAS 5-3342

Prepared for

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
GODDARD SPACE FLIGHT CENTER
Greenbelt, Maryland

ABSTRACT

WDL-TR2184 UNCLASSIFIED
PROGRAMMER'S MANUAL FOR
INTERPLANETARY ERROR PROPAGATION PROGRAM 420 pages
15 November 1963 NAS 5-3342

15732

This report discusses the subroutines that are used in the Interplanetary Error Propagation Program. A narrative description and card listing are provided for each subroutine; flow diagrams are included, if applicable. A listing of assigned locations in common is also presented.

Author

FOREWORD

This report is submitted to the National Aeronautics and Space Administration, Goddard Space Flight Center, in fulfilling the requirements of Contract NAS 5-3342.

The documentation provided by Philco WDL in support of the Interplanetary Error Propagation Program consists of the following three volumes:

- WDL-TR2184, "Programmer's Manual for Interplanetary Error Propagation Program"
- WDL-TR2185, "User's Manual for Interplanetary Error Propagation Program"
- Guidance and Control System Engineering Department Technical Report No. 4," The application of State Space Methods to Navigation Problems," by Stanley F. Schmidt

These volumes discusse the theory of the Schmidt-Kalman filter used in the program for data smoothing, the manner in which the program is used, and subroutine description and listing.

Page intentionally left blank

TABLE OF CONTENTS

Section		Page
1	INTRODUCTION	1-1
	1.1 General.	1-1
	1.2 General Program Description.	1-1
Main Chains		
	MAIN(Chain 1)	M-3
	MAIN(Chain 2)	M-13
	MAIN(Chain 3)	M-19
Subroutine Listing		
	ARKTAN	S-3
	ARKTNS	S-5
	ASINH	S-7
	BODY	S-9
	BVEC	S-15
	CHNGP	S-23
	COMPHQ	S-27
	CONST1	S-41
	CONVPI	S-51
	CORRTP	S-63
	CROSS	S-67
	(CSH)S	S-69
	DE6FN	S-71
	DOT	S-117
	EARTR	S-119
	ECLIP	S-145
	ENCKE	S-147
	ERP	S-151
	ERPT	S-153
	FINP	S-155
	FNORM	S-173

TABLE OF CONTENTS (Cont'd)

Section	Page
Subroutine Listing (Cont'd)	
GHA	S-175
GOTOB	S-179
GOTOR	S-197
GUID	S-203
HOUR	S-217
HPHT	S-221
INPUT	S-223
INTR, INTRI	S-231
INV3	S-245
INVAO	S-249
LOADO	S-251
LOADT	S-253
MASS	S-255
MATRX	S-259
MATSUB	S-261
MNA	S-267
MNAND	S-273
MONBTR	S-277
MULT	S-285
NUTAIT	S-287
OBLN	S-295
ONBTR	S-303
ORTC	S-307
OUTC	S-311
OUTDAT	S-321
OUTP	S-325
PTRAN	S-338
RETRO	S-341
ROTATE	S-347
ROTEQ	S-349
RVIN	S-355

TABLE OF CONTENTS (Cont'd)

Section	Page
Subroutine Listing (Cont'd)	
RVOUT	S-361
SDEC	S-365
SETN	S-369
SHIFTP	S-371
STEPC	S-375
TIMEC	S-391
TIMED	S-395
TRAC	S-397
TRANSH	S-403

LIST OF TABLES

Number		Page
1-1	List of Assigned Quantities in Common "C" Array.	1-3
1-2	List of Assigned Quantities in Common "S" Array.	1-6
1-3	List of Assigned Quantities in Common "IC" Array.	1-12
1-4	List of Assigned Quantities in Common "T" Array.	1-15

Page Intentionally Left Blank

PROGRAM SUBROUTINE LISTING

ARKTNS	Single Precision Arctangent
ARKTAN	Double Precision Arctangent
ASINH(X)	Function Evaluation
BODY	Calculates Accelerations Due to Perturbing Bodies
BVEC	Calculates B Vector
CHNGP	Determines when to Shift Body Center
COMPHQ	Computations for ONBTR and MONBTR Subroutines
CONST1	Array of Input Constants
CONVPI	Converts Input Covariance Matrix to 1950
CORRTP	Updates P Matrix
CROSS	Cross Product
(CSH)S	Fortran II Card Image Input Subroutine
DE6FN	Fap Integration Subroutine
DOT	Function Forming Dot Product
EARTR	Updates Covariance Matrix for Earth Based Tracking
ECLIP	Transforms Coordinates through Transformations
ENCKE	Calculates Perturbation due to Deviation from CONIC
ERP	Prints Out Ephemeris Error
ERPT	Prints Out Time of Ephemeris Error
FINP	Data Input Subroutine
FNORM	Norm of A Vector
GHA	Greenwich Hour Angle
GOTOB	Main Subroutine for Integration of Trajectories
GOTOR	Iterates to Solve Kepler's Equation
GUID	Performs Guidance Calculations
HOUR	Reads Printer Clock
HPHT	Performs Matrix Multiplication H*P*HT
INPUT	Converts Inputs to Equinox of 1950 Reference
INTR	FAP Ephemeris Subroutine
INV3	Inverts Up to a 6 By 6 Matrix
INVAO	Forms Inverse of Transistion Matrix
LOADO	Obtains Transition Matrix From T Array

LOADT	Puts Unit ICS On Perturbation Equations
MASS	Arranges Gravitational Constants of Bodies Considered
MATRX	Multiplies A*B=C or A*B*AT=C Max Dimension (10.10)
MATSUB	Error Propagation Logic Subroutine
MNA	Transformation to Selenocenic Coordinates
MNAND	Transformation for Selenocenic Velocities
MONBTR	Updates Covariance Matrix for Moon Beacons
MULT	Multiplies Two 3 by 3 Matrices
NUTAIT	Calculates Nutation Matrix
OBLN	Calculates Acceleration Due to Oblateness
ONBTR	Updates Covariance Matrix for Onboard Tracking
ORTC	Outputs Orbital Parameters
OUTC	Outputs Trajectory
OUTDAT	Outputs Calendar Date
OUTP	Outputs RMS Values of Orbital Parameters
PTRAN	Transforms P Matrix
RETRO	Performs Retro Fire
ROTATE	Calculates Transformation for Rotation About an Axis
ROTEQ	Calculates Matrix from Equinox 1950 to Mean Equinox of Date
RVIN	Transforms Coordinates From Spherical to Cartesian
RVOUT	Transforms Coordinates from Cartesian to Spherical
SDEC	Second Derivative Subroutine
SETN	Set Read and Write Tape Numbers
SHIFTP	Shifts Body Center
STEPC	Move Along Conic in Time
TIMEC	Converts Calendar Date to Days from 1950
TIMED	Converts Days Hours Min Sec to Seconds
TRAC	Tracking Station Coordinates
TRANSH	Transforms H Matrix From Date to 1950

WDL-TR2184

PROGRAMMER'S MANUAL FOR INTERPLANETARY ERROR PROPAGATION PROGRAM

SECTION 1

INTRODUCTION

1.1 GENERAL

This manual contains subroutine descriptions, logic, and FORTRAN listings for the Interplanetary Error Propagation Program. Listings of the quantities which are in the four common storage arrays (T,S,C, and IC) are also included.

1.2 GENERAL DESCRIPTION OF METHOD OF ORBIT DETERMINATION USED IN PROGRAM

The classical approach to the determination of an orbit from tracking data is the maximum likelihood, least-squares technique. This program uses a different approach to this problem. The relationships between the observed tracking data and the initial conditions (or an equivalent set of parameters) which determine the orbit are highly nonlinear. Therefore, it is assumed that a preliminary estimate of the initial conditions and, therefore, the orbit, is known. The equations are then linearized about this "nominal" orbit and an adjustment to the nominal orbit found from the differences between the observed tracking data and the predicted tracking data based on the nominal orbit. In the least-squares technique, the data is processed in parallel in the sense that the normal equations are summed over all the available tracking data before a modification to the nominal orbit is determined.

The Schmidt-Kalman technique differs from the classical least-squares approach in two ways: first, the estimate used is based upon the minimization of a risk function different from that used in the least-squares approach and, second, the data is processed serially instead of in parallel. The Schmidt-Kalman technique depends upon a linearization of the relations between the orbit parameters and the observed tracking data. Therefore, it assumes some nominal trajectory and an associated error estimate (covariance matrix) for the nominal trajectory.

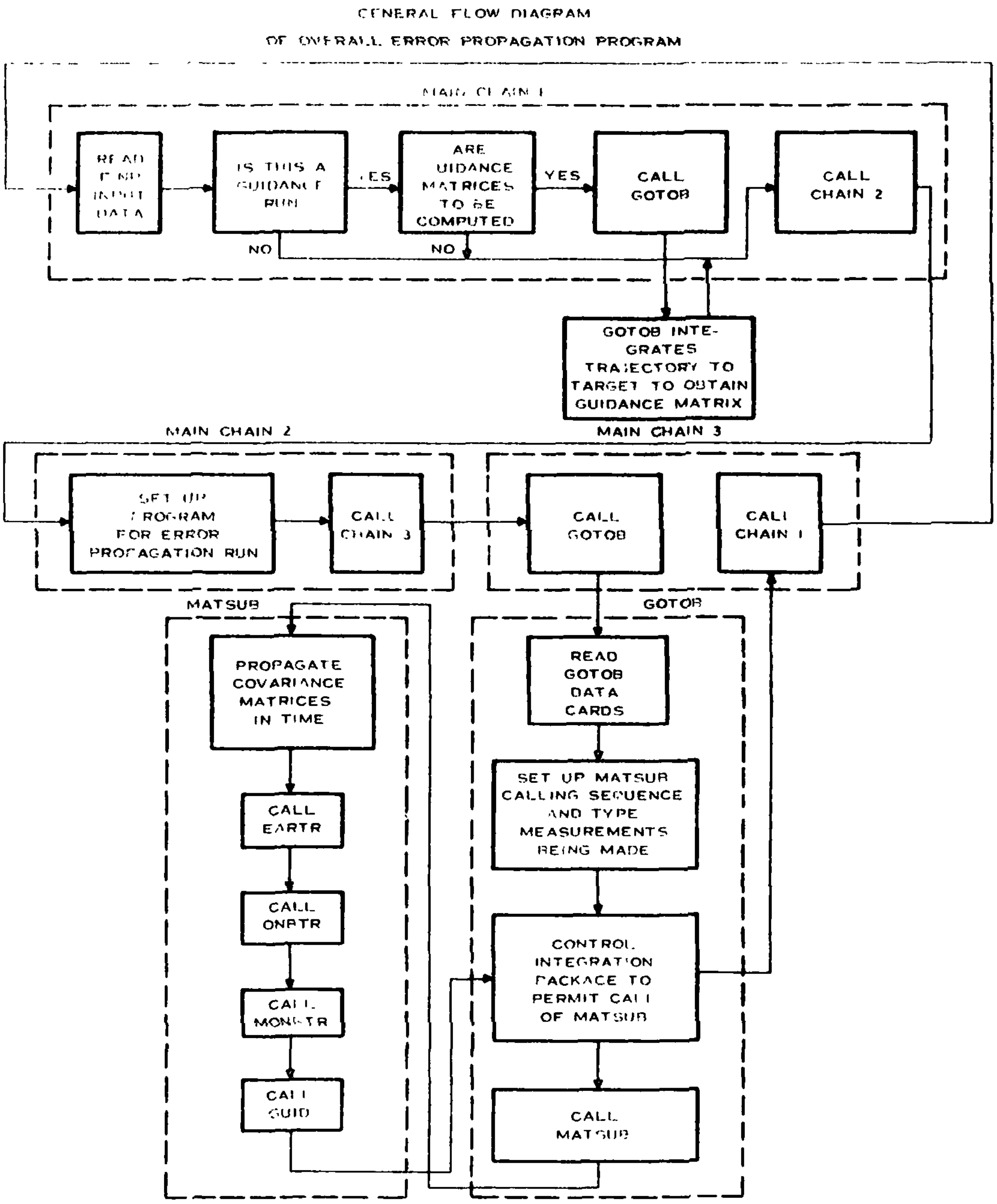

GENERAL FLOW DIAGRAM
OF OVERALL ERROR PROPAGATION PROGRAM
MAIN CHAIN 1
READ AND INPUT DATA
IS THIS A GUIDANCE RUN
YES
ARE GUIDANCE MATRICES TO BE COMPUTED
YES
CALL GOTOB
CALL CHAIN 2
NO
NO
GOTOB INTEGRATES TRAJECTORY TO TARGET TO OBTAIN GUIDANCE MATRIX
MAIN CHAIN 2
MAIN CHAIN 3
SET UP PROGRAM FOR ERROR PROPAGATION RUN
CALL CHAIN 3
CALL GOTOB
CALL CHAIN 1
MATSUB
GOTOB
PROPAGATE COVARIANCE MATRICES IN TIME
READ GOTOB DATA CARDS
SET UP MATSUB CALLING SEQUENCE AND TYPE MEASUREMENTS BEING MADE
CALL EARTR
CALL ONRTR
CONTROL INTEGRATION PACKAGE TO PERMIT CALL OF MATSUB
CALL MONSTR
CALL GUID
CALL MATSUB

Figure 1-1

The serial processing of the tracking data and associated updating of the nominal orbit occurs as follows: At the start of the time period for which tracking data are to be processed and an improved estimate of the orbit is to be obtained, it is assumed that an estimate of the orbit (a set of state variables) and an error estimate exist. Both of these estimates are updated to an instant of time when tracking data are available. Based upon the available tracking data at this time, a revised estimate of the nominal trajectory is obtained together with a revised estimate of the associated covariance matrix. These two estimates are then propagated along the orbit until additional tracking data become available. In this way, the orbit is updated and tracking data are processed serially.

1.3 GENERAL PROGRAM DESCRIPTION

This program is capable of error studies and their propagation along lunar and planetary trajectories. The design of the program is such as to facilitate the accurate and rapid evaluation of tracking systems for a given lunar or interplanetary trajectory. The program is also able to analyze the errors based on ground tracking as well as on-board tracking independently and in combination. The program has the option to calculate the errors and their projection to the target planets for a probe or a spacecraft based on the following type of measurements.

a. <u>Earth-based tracking</u>
   1. Range
   2. Range rate
   3. Azimuth
   4. Elevation
   5. Minitrack data

b. <u>On-board tracking</u>
   1. Right ascension
   2. Declination
   3. Range
   4. Range rate
   5. Range and range rate from beacons on the target planets.

The program is also capable of propagating guidance covariance matrices along the nominal trajectory. The program has the facility of using two types of guidance laws: (1) Fixed time of arrival, and (2) Variable time of arrival with the constraint that the energy with respect to the target be held constant.

The program is composed of three chains. Each link is complete in itself; that is, each contains a main program and all necessary subroutines. Control is passed from one link to the next by the CALL CHAIN statement. The program requires an ephemeris tape for operation. Figure 1-1 shows a general block diagram of the flow through the program.

The following tables show the assigned locations in "common storage":

Table 1-1:	"C" Array -----------	Floating Point Variables
Table 1-2:	"S" Array -----------	Floating Point Constants
Table 1-3:	"IC" Array ----------	Fixed Point Variables
Table 1-4:	"T" Array -----------	Storage for Integration Routine

TABLE 1-1

LIST OF ASSIGNED QUANTITIES IN COMMON "C" ARRAY

DIMENSION 1000			
CELL	FORTRAN NAME	DIMENSION	DESCRIPTION
C(10)	TW		Whole days from 1950 at last rectification
C(11)	TF		Fractional days from 1950 at last rectification
C(12)	TSECO		Time in seconds at last rectification
C(13)	TP1		Time in whole days
C(14)	TP2		Time in fractional days
C(15)	XP	(3)	X, Y, Z, equinox of 1950
C(18)	VXP	(3)	$\dot{X}$, $\dot{Y}$, $\dot{Z}$ equinox of 1950
C(21)	AE	(3)	Acceleration due to deviation from conic
C(24)	AO	(3)	Acceleration due to oblateness
C(27)	CA	(3)	Acceleration due to perturbing bodies
C(30)	TSEC		Seconds from start - zero reference time
C(33)	X	(3)	Reference conic X, Y, Z
C(36)	VX	(3)	Reference conic $\dot{X}$, $\dot{Y}$, $\dot{Z}$
C(46)	XSO	(3)	Initial position for STEPC conic
C(49)	VXSO	(3)	Initial velocity for STEPC conic
C(55)	μ		Gravity constant of central body
C(56)	VKB	(6)	Gravitational constants of bodies used
C(62)	PO	(22)	Ephemeris positions of bodies
C(84)	VE	(22)	Ephemeris velocities of bodies
C(106)	RBO	(6)	Distances to bodies

DIMENSION 1000			
CELL	FORTRAN NAME	DIMENSION	DESCRIPTION
C(112)	RBOP	(6)	Distances to bodies
C(120)	EN	(3,3)	Nutation matrix
C(129)	EA	(3,3)	Transformation from 1950 to mean equinox of date
C(138)	AN	(3,3)	Transformation from 1950 to true equator of date
C(147)	ECL	(3,3)	Transformation from 1950 to ecliptic of date
C(156)	ECX	(3)	Position: ecliptic coordinates
C(159)	ECV	(3)	Velocity: ecliptic coordinates
C(162)	BET		Angle from line of nodes to radius vector
C(163)	THT		True anomaly
C(171)	BP	(3,3)	Acceleration partials for use in variational equations
C(180)	SMA		Semi-Major Axis
C(181)	ECC		Eccentricity
C(182)	RCA		Radius of closest approach
C(183)	OINC		Orbital inclination
C(184)	OMG		Longitude of ascending node
C(185)	BEP		Argument of periapsis
C(186)	XED	(3)	Position: equator date
C(189)	VED	(3)	Velocity: equator date
C(192)	RCV	(3)	$\vec{R} \times \vec{v}$
C(195)	RCV2		$\lvert\vec{R} \times \vec{v}\rvert^2$
C(196)	R2		R^2

DIMENSION 1000			
CELL	FORTRAN NAME	DIMENSION	DESCRIPTION
C(197)	V^2		V^2
C(198)	RDV		$\vec{R} \cdot \vec{V}$
C(199)	C3		$C_3 = V^2 - \frac{2\mu}{R}$
C(460)	PFTA	(6,6)	Deviation From Nominal Covariance Matrix
C(568)	TRANS	(3,6)	Guidance partial matrix being used on runs
C(622)	TLAST	(4,6)	Delta time since last observation in ONBTR
C(647)	TOUT		Time to output for MATSUB
C(648)	TGUIDE		Time to make guidance correction
C(649)	TSECP		LAST TIME through MATSUB
C(650)	TPRINT		Output print interval
C(651)	TSTART		Start time
C(652)	P	(6,6)	Knowledge of state covariance matrix
C(752)	AOS	(6,6)	Transition matrix from start to present time
C(788)	STPARS	(3,3)	Station partials MONBTR
C(797)	CIOMP		Compute guidance matrices
C(800)	STPARD	(3,3)	Station partials MONBTR
C(888) C(900)			Constants used in COMPHQ
C(973)	OUTPUT	(6)	Data for output purposes

TABLE 1-2

LIST OF ASSIGNED QUANTITIES IN COMMON "S" ARRAY

DIMENSION 1000			
CELL	FORTRAN NAME	DIMENSION	DESCRIPTION
S(1)			Earth Planetary Mass
S(2)			Moon Planetary Mass
S(3)			Sun Planetary Mass
S(4)			Venus Planetary Mass
S(5)			Mars Planetary Mass
S(6)			Jupiter Planetary Mass
S(7)			SPARE
S(8)			SPARE
S(9)			SPARE
S(10) S(19)			Earth Constants
S(20) S(29)			Moon Constants
S(30) S(39)			Sun Constants
S(40) S(49)			Venus Constants
S(50) S(59)			Mars Constants
S(60) S(69)			Jupiter Constants
S(70)			86400. Conversion factor days to seconds
S(71)			1/86400. Conversion factor seconds to days

DIMENSION 1000			
CELL	FORTRAN NAME	DIMENSION	DESCRIPTION
S(72)			.017453296 Conversion factor for degrees to radians
S(73)			57.29578 Conversion factor for radians to degrees
S(80)	AT		Launch latitude - degrees
S(81)	ON		Launch longitude - degrees
S(82)	AL		Launch altitude - km.
S(83)	TLTI		Time from launch to injection, days hours. min sec
S(84)	FAZ		Firing azimuth
S(110)	TSTOP		Time to stop calling MATSUB
S(111)	TIM		Type of input
S(112)	DATE		Start date
S(113)	FDATE		Fractional start date
S(114)	CIBDY		Central body
S(115)	X	(3)	Initial position
S(118)	VX	(3)	Initial velocity
S(121)	TIBDY		Target Body
S(122)	OUTTP		Type of output
S(124)	SKTB		Type of stop

EARTH BASED TRACKING DATA

N = 0 - 19

CELL	DESCRIPTION	UNITS
S(125)+N*15	Variance of range	KM^2
S(126)+N*15	Variance of azimuth	RAD^2
S(127)+N*15	Variance of elevation	RAD^2
S(128)+N*15	Variance of range rate	KM^2/SEC^2
S(129)+N*15	Variance of latitude	RAD^2
S(130)+N*15	Variance of longitude	RAD^2
S(131)+N*15	Variance of altitude	KM^2
S(132)+N*15	Variance of azimuth biases	RAD^2
S(133)+N*15	Variance of elevation biases	RAD^2
S(134)+N*15	Latitude of station	DEG
S(135)+N*15	Longitude of station	DEC
S(136)+N*15	Altitude of station	KM
S(137)+N*15	Station name	
S(138)+N*15	Period of observation	SEC
S(139)+N*15	Variance of time bias	SEC^2
S(425)	Velocity of light error	KM^2/SEC^2

ON-BOARD TRACKING DATA

S(426)	Variance of range	KM^2
S(427)	Variance of right ascension	RAD^2
S(428)	Variance of declination	RAD^2
S(429)	Variance of range rate	KM^2/SEC^2
S(430)	Variance of range bias	KM^2
S(431)	Variance of right ascension bias	RAD^2

CELL	DESCRIPTION	UNITS
S(432)	Variance of declination bias	RAD^2
S(433)	Variance of range rate bias	KM^2/SEC^2
S(434)	Variance of clock time bias	SEC^2
S(435)	Spare	
S(436)	Spare	

N = 0 - 5 EARTH, MOON,SUN, VENUS, MARS, JUPITER

CELL	FORTRAN NAME	DIMENSION	DESCRIPTION	UNITS
S(437)+4*N			Period of range observation	SEC
S(438)+4*N			Period of right ascension Observation	SEC
S(439)+4*N			Period of declination Observation	SEC
S(440)+4*N			Period of range rate Observation	SEC
S(471)			Shutoff Error (Rocket)	$(\%)^2$
S(473)			Pointing Error (Motor)	$(RAD)^2$
S(475)			Type of guidance decision 0-Time + Ratio	
S(476)	TGUID	(6)	Times for guidance correction	
S(483)	GDCOR		Ratio for making guidance correction	
S(484)	DMON		Error in monitoring guidance correction	$(\%)^2$
S(485)	ENKE		Correction criteria for deivation from reference conic	

CELL	FORTRAN NAME	DIMENSION	DESCRIPTION	UNITS
			MOON BEACON TRACKING DATA N = 0 - 9	
S(500)+N*15			Variance of range	KM^2
S(501)+N*15			Variance of right ascension	RAD^2
S(502)+N*15			Variance of declination	RAD^2
S(503)+N*15			Variance of range rate	KM^2/SEC^2
S(504)+N*15			Variance of range bias	KM^2
S(505)+N*15			Variance of right ascension bias	RAD^2
S(506)+N*15			Variance of declination bias	RAD^2
S(507)+N*15			Variance range rate bias	KM^2/SEC^2
S(508)+N*15			Variance clock bias	SEC^2
S(509)+N*15			Latitude of station	DEG
S(510)+N*15			Longitude of station	DEG
S(511)+N*15			Altitude of station	KM
S(512)+N*15			Station name	
S(513)+N*15			Period of observation	SEC
S(514)+N*15			SPARE	
S(650)	PARI	(21)	Initial deviation from nominal covariance matrix	
S(671)	PI	(21)	Initial knowledge of state covariance matrix	
S(720)	PUPIN		Type of coordinates for PI	
S(721)	PARIN		Type of coordinates for PARI	
S(722)	FTA	(3,6)	Guidance transition matrix fixed time arrival	

CELL	FORTRAN NAME	DIMENSION	DESCRIPTION	UNITS
S(740)	CTE	(3,6)	Guidance transition matrix constant energy WRT* target	

MOON BEACON DATA

N = 0 - 9

S(758)+N*3			Variance of station latitude	RAD^2
S(759)+N*3			Variance of station longitude	RAD^2
S(760)+N*3			Variance of station altitude	KM^2

EARTH BASED TRACKING DATA

N = 0 - 19

S(788)+N*5			Tracking Station horizon	RAD
S(789)+N*5			Variance of M dircos	
S(790)+N*5			Variance of M dircos bias	
S(791)+N*5			Variance of L dircos	
S(792)+N*5			Variance of L dircos bias	
S(888)	RETR		Key to call subroutine RETRO	
S(900)	RAD	(6)	Radius to bodies	
			* with respect to	

TABLE 1-3

LIST OF ASSIGNED QUANTITIES IN COMMON "IC" ARRAY

DIMENSION 300			
CELL	FORTRAN NAME	DIMENSION	DESCRIPTION
IC(1)	N		# of sets of equations
IC(2)	IOR		body center
IC(3)	NOR		new body center
IC(4)	KSTP		type stop
IC(5)	KTYPE		switch indicating type stop
IC(6)	ITARG		target body
IC(7)	KOUT		type of output
IC(8)	IMMSUB		key to test if MATSUB is to be called
IC(10)			+ if tracking station 1 considered
IC(11)			+ if tracking station 2 considered
IC(12)			+ if tracking station 3 considered
. . .			
IC(29)			+ if station 20 considered
IC(30)			+ range, 0 no range (Station # 1)
IC(31)			+ AZ-EL, 0 no AZ-EL (Station # 1)
IC(32)			+ dircos, 0 no dircos (Station # 1)
IC(33)			+ range rate, 0 no range rate (Station # 1)
. . .			

CELL	FORTRAN NAME	DIMENSION	DESCRIPTION	
IC(106)			+ range, 0 no range	Station # 20
IC(107)			+ AZ-EL, 0 no AZ-EL	
IC(108)			+ dircos 0 no dircos	
IC(109)			+ range rate, 0 no range rate	
IC(110)			Earth observed + yes, 0 no	
IC(111)			Moon observed, + yes, 0 no	
IC(112)			Sun observed, + yes, 0 no	
IC(113)			Venus observed, + yes, 0 no	
IC(114)			Mars observed, + yes, 0 no	
IC(115)			Jupiter observed, + yes, 0 no	
IC(116)			range, + yes, 0 no	Earth on-board measurements
IC(117)			rt. ascension, + yes, 0 no	
IC(118)			declination, + yes, 0 no	
IC(119)			range rate, + yes, 0 no	
⋮				
IC(136)			range, + yes, 0 no	Jupiter on-board measurements
IC(137)			rt. ascension, + yes, 0 no	
IC(138)			declination, + yes, 0 no	
IC(139)			range rate, + yes, 0 no	
IC(140)			$\frac{+}{o}$ Moon beacon station 1	10 Moon beacon stations
⋮				
IC(149)			$\frac{+}{o}$ Moon beacon station 10	

CELL	FORTRAN NAME	DIMENSION	DESCRIPTION
IC(150)			$\underset{o}{+}$ range
IC(151)			$\underset{o}{+}$ right ascension station
IC(152)			$\underset{o}{+}$ declination one
IC(153)			$\underset{o}{+}$ range rate
. . . .			
IC(186)			$\underset{o}{+}$ range
IC(187)			$\underset{o}{+}$ right ascension station
IC(188)			$\underset{o}{+}$ declination ten
IC(189)			$\underset{o}{+}$ range rate
IC(190)	LSTAT		Key for earth based tracking
IC(191)	LONB		Key for on-board tracking
IC(192)	LMB		Key for moon beacon tracking
IC(193)	IGUID		Positive if guidance run
IC(194)	ISTAT	(20)	Keep account which of 20 trackers saw vehicle last time in MATSUB
IC(214)	NOUT		Output indicator
IC(215)	PUPIN		type of P matrix input
IC(216)	PARIN		type of PAR matrix input
IC(217)	IGDTP		type of guidance
IC(218)	IGDKEY		guidance key set in main
IC(219)	IGD		number of guidance corrections
IC(220)	IMOONB	(20)	Keep account which of 10 moon beacons saw vehicle last time in MATSUB
IC(231)	NSTEP		Key for STEPC

TABLE 1-4

LIST OF ASSIGNED QUANTITIES IN COMMON "T" ARRAY

This is the common storage block for the integration package. See the subroutine writeup for DE6FN for definitions of the quantities in storage.

DIMENSION 1360				
CELL	DESCRIPTION			
T(1)	N	No. of Equations	initial conditions which must be supplied	
T(2)	X			
T(3)	H	Step Size		
T(4)	y_i	$i = 1,N$		
T(N+4)	y_i'	$i = 1,N$		
T(2N+4)	y_i''	$i = 1,N$	second derivatives stored by V subroutine	
T(3N+4)	y_i'	$i = 1,N$	most significant	
T(4N+4)	y_i'	$i = 1,N$	least significant	
T(5N+4)	$\Delta y_i'$	$i = 1,N$		
T(6N+4)	$\Delta y_i'$	$i = 1,N$		
T(7N+4)	y_i	$i = 1,N$	most significant	
T(8N+4)	y_i	$i = 1,N$	least significant	
T(9N+4)	y_{i3}	$i = 1,N$	saved for central difference equations	
T(10N+4)	y_{i3}'	$i = 1,N$		

DIMENSION 1360	
CELL	DESCRIPTION
T(11N+4)	y''_{i0} i = 1,N
T(12N+4)	y''_{i1} i = 1,N
T(13N+4)	y''_{i2} i = 1,N
T(14N+4)	y''_{i3} i = 1,N
T(15N+4)	y''_{i4} i = 1,N
T(16N+4)	y''_{i5} i = 1,N
T(17N+4)	y''_{i6} i = 1,N
	saved from 6R, R-K steps. Used to form difference tables in an attempt to proceed to a cowell step.
T(18N+4)	used only if coordinates changed.
T(19N+4) . . . . . . T(30N-8)	11 N storages. N sets of differences saved for next cowell step.

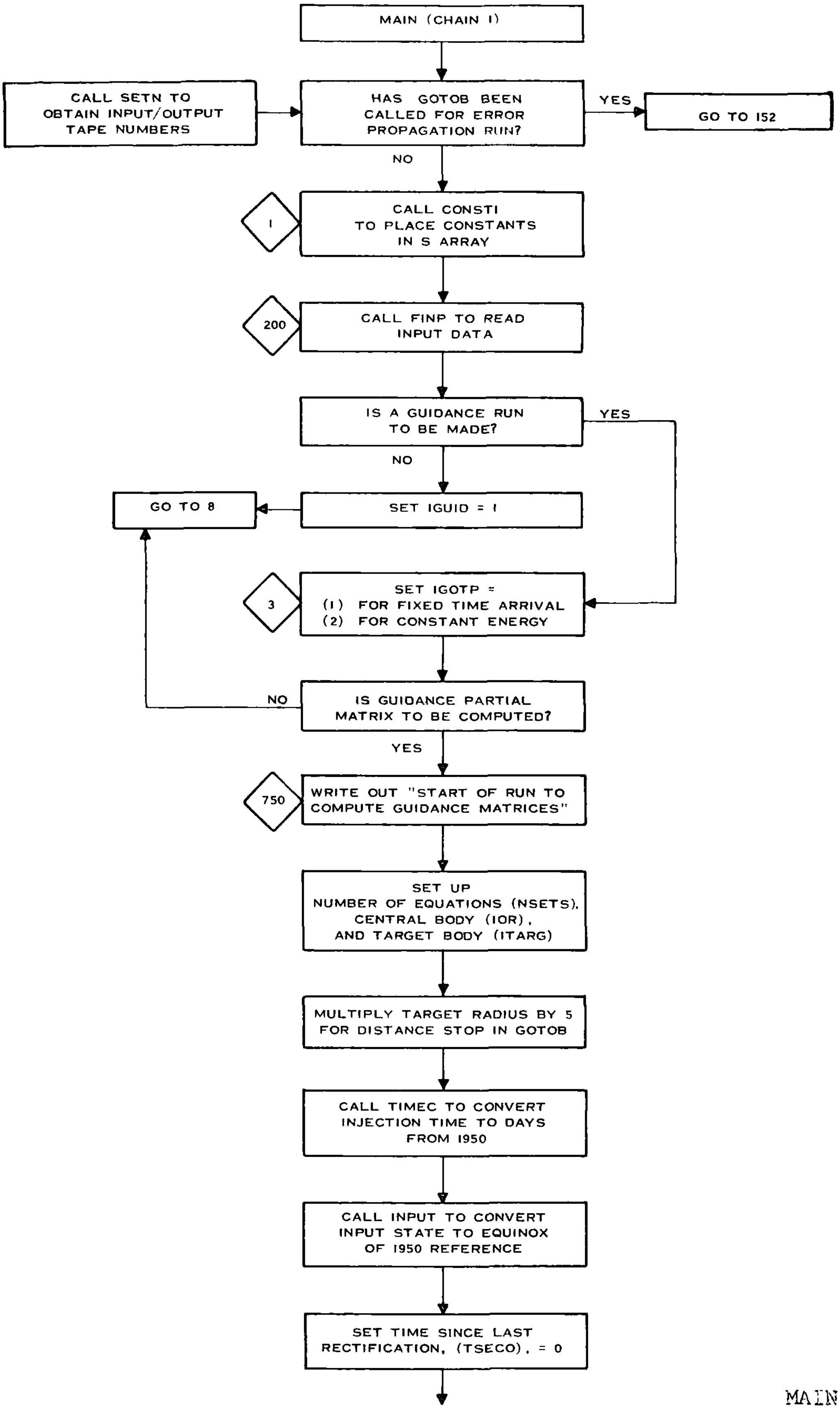
MAIN CHAINS

Page Intentionally Left Blank

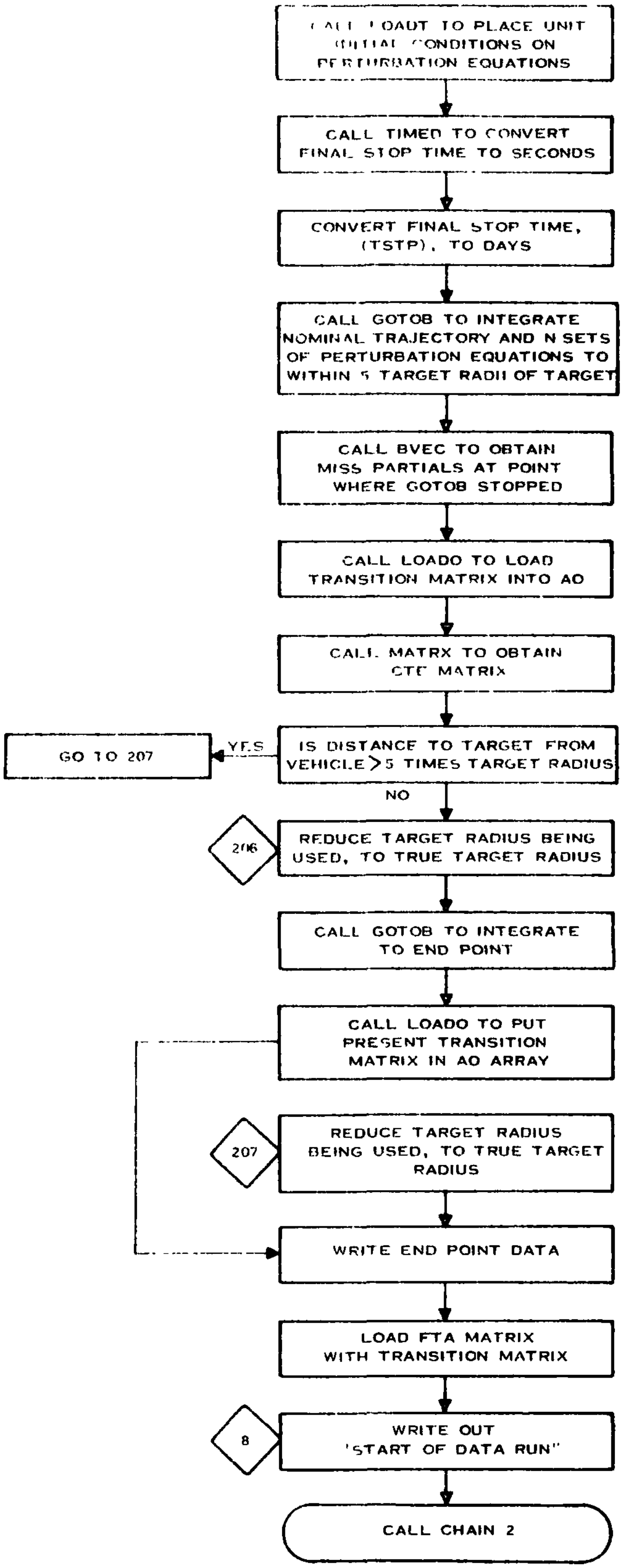
Subroutine: MAIN (Chain 1)

Purpose: Set up program to perform guidance runs to obtain transition and partial matrices. Call Chain 2 for error propagation runs and write out end point data after program returns from Chain 3.

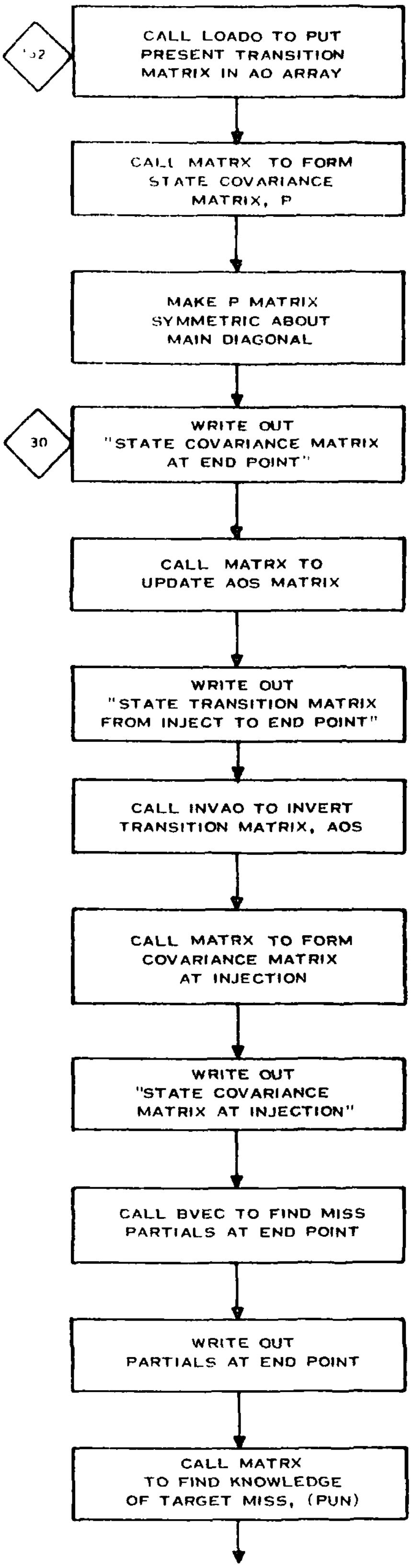
Common storages used or required:	T, S, C, IC
Subroutines required:	BVEC, CHAIN, CONST1, FINP, GOTOB, INPUT, INVAO, LOADO, LOADT, MATRX, RETRO, TIMEC, TIMED
Functions required:	None
Approximate number of storages required:	793 DEC

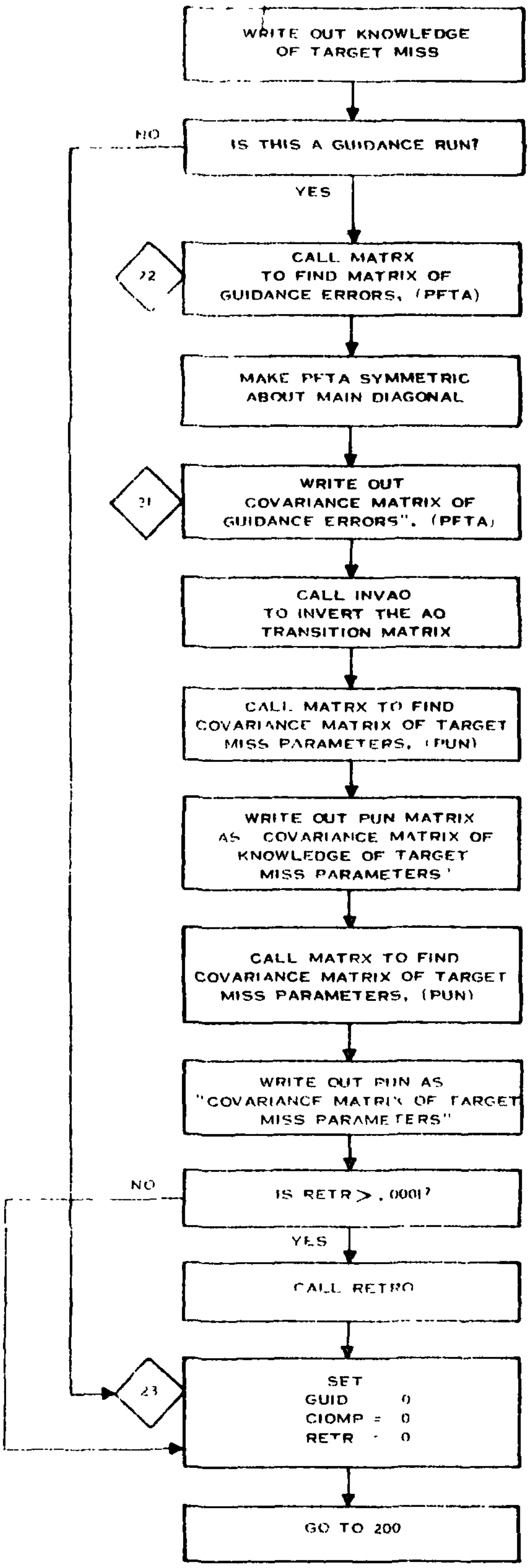


MAIN (CHAIN-1) -2



CALL LOADUT TO PLACE UNIT INITIAL CONDITIONS ON PERTURBATION EQUATIONS
CALL TIMED TO CONVERT FINAL STOP TIME TO SECONDS
CONVERT FINAL STOP TIME, (TSTP), TO DAYS
CALL GOTOB TO INTEGRATE NOMINAL TRAJECTORY AND N SETS OF PERTURBATION EQUATIONS TO WITHIN 5 TARGET RADII OF TARGET
CALL BVEC TO OBTAIN MISS PARTIALS AT POINT WHERE GOTOB STOPPED
CALL LOADO TO LOAD TRANSITION MATRIX INTO AO
CALL MATRX TO OBTAIN CTE MATRIX
IS DISTANCE TO TARGET FROM VEHICLE > 5 TIMES TARGET RADIUS
YES
GO TO 207
NO
206
REDUCE TARGET RADIUS BEING USED, TO TRUE TARGET RADIUS
CALL GOTOB TO INTEGRATE TO END POINT
CALL LOADO TO PUT PRESENT TRANSITION MATRIX IN AO ARRAY
207
REDUCE TARGET RADIUS BEING USED, TO TRUE TARGET RADIUS
WRITE END POINT DATA
LOAD FTA MATRIX WITH TRANSITION MATRIX
8
WRITE OUT 'START OF DATA RUN"
CALL CHAIN 2


32
CALL LOADO TO PUT PRESENT TRANSITION MATRIX IN AO ARRAY
CALL MATRX TO FORM STATE COVARIANCE MATRIX, P
MAKE P MATRIX SYMMETRIC ABOUT MAIN DIAGONAL
30
WRITE OUT "STATE COVARIANCE MATRIX AT END POINT"
CALL MATRX TO UPDATE AOS MATRIX
WRITE OUT "STATE TRANSITION MATRIX FROM INJECT TO END POINT"
CALL INVAO TO INVERT TRANSITION MATRIX, AOS
CALL MATRX TO FORM COVARIANCE MATRIX AT INJECTION
WRITE OUT "STATE COVARIANCE MATRIX AT INJECTION"
CALL BVEC TO FIND MISS PARTIALS AT END POINT
WRITE OUT PARTIALS AT END POINT
CALL MATRX TO FIND KNOWLEDGE OF TARGET MISS, (PUN)



WRITE OUT KNOWLEDGE OF TARGET MISS
IS THIS A GUIDANCE RUN?
NO
YES
22
CALL MATRX TO FIND MATRIX OF GUIDANCE ERRORS, (PETA)
MAKE PETA SYMMETRIC ABOUT MAIN DIAGONAL
21
WRITE OUT COVARIANCE MATRIX OF GUIDANCE ERRORS". (PETA)
CALL INVAO TO INVERT THE AO TRANSITION MATRIX
CALL MATRX TO FIND COVARIANCE MATRIX OF TARGET MISS PARAMETERS, (PUN)
WRITE OUT PUN MATRIX AS COVARIANCE MATRIX OF KNOWLEDGE OF TARGET MISS PARAMETERS'
CALL MATRX TO FIND COVARIANCE MATRIX OF TARGET MISS PARAMETERS, (PUN)
WRITE OUT PUN AS "COVARIANCE MATRIX OF TARGET MISS PARAMETERS"
IS RETR > .0001?
NO
YES
CALL RETRO
23
SET
GUID 0
CIOMP = 0
RETR = 0
GO TO 200

MAIN (CHAIN 1)

```
*      LABEL                                                                MAI1
CEC20D2   MAIN PROGRAM                                                      MAI10000
C       ERROR PROPAGATION                                                   MAI10010
C       ALONG INTERPLANETARY TRAJECTORIES                                   MAI10015
C       REQUIRES FOLLOWING SUBROUTINES AND FUNCTIONS                        MAI10020
C       ARKTNS   SINGLE PRECISION ARCTANGENT                                MAI10030
C       ARKTAN   DOUBLE PRECISION ARCTANGENT                                MAI10040
C       ASINH(X) FUNCTION EVALUATION                                        MAI10050
C       BODY     CALCULATES ACCELERATIONS DUE TO PERTURBING BODIES          MAI10060
C       BVEC     CALCULATES B VECTOR                                        MAI10070
C       CHNGP    DETERMINES WHEN TO SHIFT BODY CENTER                       MAI10090
C       COMPHQ   COMPUTATIONS FOR ONBTR AND MONBTR SUBROUTINES              MAI10095
C       CONST1   ARRAY OF INPUT CONSTANTS                                   MAI10100
C       CONVPI   CONVERTS INPUT COVARIENCE MATRIX TO 1950                   MAI10110
C       CORRTP   UPDATES P MATRIX                                           MAI10020
C       CROSS    CROSS PRODUCT                                              MAI10130
C       DOT      FUNCTION FORMING DOT PRODUCT                               MAI10140
C       DE6FN    FAP  INTEGRATION SUBROUTINE                                MAI10160
C       EARTR    UPDATES COVARIANCE MATRIX FOR EARTH BASED TRACKING         MAI10170
C       ECLIP    TRANSFORMS COORDINATES THROUGH TRANSFORMATIONS             MAI10180
C       ENCKE    CALCULATES PERTURBATION DUE TO DEVIATION FROM CONIC        MAI10190
C       ERP      PRINTS OUT EPHEMERIS ERROR                                 MAI10195
C       ERPT     PRINTS OUT TIME OF EPHEMERIS ERROR                         MAI10197
C       FINP     DATA INPUT SUBROUTINE                                      MAI10200
C       FNORM    NORM OF A VECTOR                                           MAI10210
C       GHA      GREENWICH HOUR ANGLE                                       MAI10220
C       GOTOB    MAIN SUBROUTINE FOR INTEGRATION OF TRAJECTORIES            MAI10230
C       GOTOR    ITERATES TO SOVE KEPLERS EQUATION                          MAI10235
C       GUID     PERFORMS GUIDANCE CALCULATIONS                             MAI10240
C       HPHT     PERFORMS MATRIX MULTIPLICATION H*P*HT                      MAI10245
C       INPUT    CONVERTS INPUTS TO EQUINOX OF 1950 REFERENCE               MAI10250
C       INTR     FAP EPHEMERIS SUBROUTINE                                   MAI10260
C       INVAO    FORMS INVERSE OF TRANSITION MATRIX                         MAI10270
C       INV3     INVERTS UP TO A 6 BY 6 MATRIX                              MAI10280
C       LOADO    OBTAINS TRANSITION MATRIX FROM T ARRAY                     MAI10290
C       LOADT    PUTS UNIT ICS ON PERTURBATION EQUATIONS                    MAI10300
C       MASS     ARRANGES GRAVITATIONAL CONSTANTS OF BODIES CONSIDERED      MAI10310
C       MATRX    MULTIPLIES A*B=C OR A*B*AT=C MAX DIMENSION(10,10)          MAI10320
C       MATSUB   ERROR PROPOGATION LOGIC SUBROUTINE                         MAI10330
C       MNA      TRANSFORMATION TO SELENOCENTIC COORDINATES                 MAI10340
C       MNAND    TRANSFORMATION FOR SELENOCENTRIC VELOCITIES                MAI10345
C       MONBTR   UPDATES COVARIANCE MATRIX FOR MOON BEACONS                 MAI10350
C       MULT     MULTIPLIES TWO 3 BY 3 MATRICES                             MAI10370
C       NUTAIT   CALCULATES NUTATION MATRIX                                 MAI10380
C       OBLIN    CALCULATES ACCELERATION DUE TO OBLATENESS                  MAI10390
C       ONBTR    UPDATES COVARIANCE MATRIX FOR ONBOARD TRACKING             MAI10400
C       ORTC     OUTPUTS ORBITAL PARAMETERS                                 MAI10420
C       OUTC     OUTPUTS  TRAJECTORY                                        MAI10440
C       OUTDAT   OUTPUTS CALENDAR DATE                                      MAI10430
C       OUTP     OUTPUTS RMS VALUES OF ORBITAL PARAMETERS                   MAI10450
C       PTRAN    TRANSFORMS P MATRIX                                        MAI10452
```

AIN (CHAIN 1) (Cont'd)

```
C       RETRO    PERFORMS RETRO FIRE                                          MAI10455
C       ROTATE   CALCULATES TRANSFORMATION FOR ROTATION ABOUT AN AXIS         MAI10460
C       ROTEQ    CALCULATES MATRIX FROM EQUINOX 1950 TO MEAN EQUINOX          MAI10465
C                OF DATE                                                      MAI10468
C       RVIN     TRANSFORMS COORDINATES FROM SPHERICAL TO CARTESIAN           MAI10470
C       RVOUT    TRANSFORMS COORDINATES FROM CARTESIAN TO SPHERICAL           MAI10480
C       SDEC     SECOND DERIVATIVE SUBROUTINE                                 MAI10530
C       SETN     SET READ AND WRITE TAPE NUMBERS                              MAI10535
C       SHIFTP   SHIFTS BODY CENTER                                           MAI10540
C       STEPC    MOVE ALONG CONIC IN TIME                                     MAI10545
C       TIMEC    CONVERTS CALENDAR DATE TO DAYS FROM 1950                     MAI10550
C       TIMED    CONVERTS DAYS HOURS,MIN SEC TO SECONDS                       MAI10560
C       TRAC     TRACKING STATION COORDIATES                                  MAI10570
C       TRANSH   TRANSFORMS H MATRIX FROM DATE TO 1950                        MAI10580
      COMMON T,S,C,IC                                                         MAI10600
      DIMENSION T(1360),S(1000),C(1000),IC(300),X(3)                          MAI10610
     1,VX(3),XP(3),VXP(3),AN(3,3),UM(6),BV(4),PBV(3,6)                        MAI10620
     2,AO(6,6),PARI(21),PI(21),ISTAT(20),FTA(3,6)                             MAI10630
     3,TRANS(3,6)                                                             MAI10640
     4,PFTA(6,6),P(6,6),TGUID(6)                                              MAI10650
      DIMENSION AOS(6,6),RAD(6), RBOP(6),CTE(3,6),AOI(6,6),PUN(3,3)           MAI10660
      EQUIVALENCE(T,TDUM),(S,SDUM),(C,CDUM),(IC,IDUM),                        MAI10670
     1(IC(8),IMMSUB),(S(112),DATE),(S(113),FDATE),(S(115),X),(S(118),VX)MAI10680
     2,(IC(193),IGUID),(S(110),TSTOP),(IC(2),IOR),(S(111),TIM)                MAI10690
     3,(IC(6),ITARG),(IC(4),KSTP),(IC(1),NSETS),(C(10),TW),(C(11),TF)         MAI10700
     4,(C(138),AN),(C(12),TSECO),(S(71),STD),(S(1),UM),(C(15),XP)             MAI10710
     5,(C(18),VXP),(C(13),TP1),(C(14),TP2),(S(671),PI),(S(650),PARI)          MAI10720
     6,(IC(194),ISTAT),(IC(7),KOUT),(C(649),TSECP),(S(122),OUTTP)             MAI10730
     7,(C(460),PFTA),(C(652),P)                                               MAI10740
     8,(S(114),CIBDY),(S(121),TIBDY),(S(722),FTA),(S(124),SKTB)               MAI10750
     9,(S(720),PUPIN),(S(721),PARIN)                                          MAI10760
      EQUIVALENCE (C(752),AOS),(S(900),RAD),(C(112),RBOP),(C(30),TSEC)        MAI10770
     1,(S(740),CTE),(C(648),TGUIDE),(C(850),AOI),(C(850),PUN)                 MAI10780
      EQUIVALENCE (S(476),TGUID),(IC(219),IGD)                                MAI10790
     1,(IC(218),IGDKY),(IC(217),IGDTP),(C(568),TRANS)                         MAI10800
      EQUIVALENCE (T(1358),TSTP),(C(797),COMP),(S(888),RETR)                  MAI10810
      CALL SETN(NIN,NUTS)                                                     MAI10811
      IF(T(1359)-1359.)1,152,1                                                MAI10820
 1    CALL CONST1                                                             MAI10830
C     THE CONSTS OF S ARRAY ARE READ IN                                       MAI10840
 200  CONTINUE                                                                MAI10850
      WRITE OUTPUT TAPE NUTS,700                                              MAI10851
 700  FORMAT(26H1 START OF FINP INPUT DATA)                                   MAI10855
      CALL FINP(22,S,TIM,CIBDY,TIBDY,GUID,SKTB,CIOMP                          MAI10860
     1,TSTOP,X,VX,DATE,FDATE,FTA,PI,PARI,TGUID,GID,PUPIN,PARIN,CTE,           MAI10865
     2RETR,OUTTP,                                                         132HMAI10870
     2S      TIM    CIBDY TIBDY GUID  SKTB  CIOMP TSTOP X      VX    DATE  MAI10880
     3FDATE FTA    PI     PARI  TGUID GID   PUPIN PARIN CTE    RETR  OUTTP MAI10885
     4)                                                                       MAI10890
C      GUID IS SET POSITIVE IF GUIDANCE IS CONSIDERED                         MAI10900
      IF(GUID) 2,2,3                                                          MAI10910
```

MAIN (CHAIN 1) (Cont'd)

```
    2  CONTINUE
       IGUID=1
       GO TO 8
    3  CONTINUE
       IGD=GID
       IGDTP = GUID
       IF(IGDTP-3) 204,204,205
  205  CONTINUE
       IGDTP=2
  204  CONTINUE
       IGUID=2
C       CIOMP IS SET POSITIVE IF MATRICES ARE TO BE COMPUTED
       IF(CIOMP) 8,8,4
    4  CONTINUE
       COMP=CIOMP
       WRITE OUTPUT TAPE NUTS,750
  750  FORMAT(43H1START OF RUN TO COMPUTE GUIDANCE MATRICIES)
       NSETS=7
       IOR=S(114)
       ITARG=S(121)
       RAD(ITARG)=5.*RAD(ITARG)
       KIN=S(111)
       CALL TIMEC(DATE,FDATE,TW,TF)
       CALL INPUT(KIN,XP,VXP,TW,TF,AN,X,VX)
       TSECO =0.
       KSTP=1
       CALL LOADT
       CALL TIMED(TSTOP,TSTPS)
       TSTP=TSTPS*STD
       IMMSUB=1
       CALL GOTOB(TSTP)
       CALL BVEC(IOR,ITARG,XP,VXP,BV,PBV,TP1,TP2,UM)
       CALL LOADO(AO)
       CALL MATRX(PBV,AO,CTE,3,6,6,0)
       IF(RBOP(ITARG)-RAD(ITARG))206,206,207
  206  CONTINUE
       RAD(ITARG)=RAD(ITARG)/5.
       TSECO=TSEC
       CALL GOTOB(TSTP)
       CALL LOADO(AO)
       GO TO 208
  207  CONTINUE
       RAD(ITARG)=RAD(ITARG)/5.
  208  CONTINUE
       WRITE OUTPUT TAPE NUTS,706,((AO(I,J),J=1,6),I=1,6)
  706  FORMAT(31H0TRANSITION MATRIX TO END POINT//(6E17.8))
       WRITE OUTPUT TAPE NUTS,707,((CIE(I,J),J=1,6),I=1,3)
  707  FORMAT(49H0B VECTOR TRANSITION MATRIX, BY ROWS B.T,B.R,VINF//
      1(6E17.8))
       DO 6 I=1,3
       DO 6 J=1,6
```

```
      FTA(I,J)=AO(I,J)                                                  MAI11430
    6 CONTINUE                                                          MAI11440
    8 CONTINUE                                                          MAI11450
      WRITE OUTPUT TAPE NUTS,751                                        MAI11460
  751 FORMAT(18H1START OF DATA RUN)                                     MAI11470
      CALL CHAIN(2,B3)                                                  MAI11480
  152 CALL LOADO(AO)                                                    MAI11490
C     AO IS TRANSITION MATRIX FROM LAST TIME IN MATSUB                  MAI11500
      CALL MATRX(AO,P,P,6,6,6,1)                                        MAI11510
      DO 30 I=1,6                                                       MAI11511
      DO 30 J=I,6                                                       MAI11512
      P(I,J)=(P(I,J)+P(J,I))/2.                                         MAI11514
      P(J,I)=P(I,J)                                                     MAI11516
   30 CONTINUE                                                          MAI11518
C     P IS STATE COVARIACE MATRIX AT END POINT                          MAI11520
      WRITE OUTPUT TAPE NUTS,800,((P(I,J),J=1,6),I=1,6)                 MAI11530
  800 FORMAT(38H0 STATE COVARIANCE MATRIX AT END POINT                  MAI11540
     1/(6E17.8))                                                        MAI11550
      CALL MATRX(AO,AOS,AOS,6,6,6,0)                                    MAI11560
C     AOS IS TRANSITION MATRIX FROM INJECTION TO END POINT              MAI11570
      WRITE OUTPUT TAPE NUTS,801,((AOS(I,J),J=1,6)I=1,6)                MAI11580
  801 FORMAT(50H0 STATE TRANSITION MATRIX FROM INJECT TO END POINT      MAI11590
     1/(6E17.8))                                                        MAI11600
      CALL INVAO(AOS,AOI)                                               MAI11610
C     FORMS INVERSE OF AOS                                              MAI11620
      CALL MATRX(AOI,P,ACS,6,6,6,1)                                     MAI11630
      WRITE OUTPUT TAPE NUTS,802,AOS                                    MAI11640
  802 FORMAT(38H0 STATE COVARIANCE MATRIX AT INJECTION                  MAI11650
     1/(6E17.8))                                                        MAI11660
      CALL BVEC(IOR,ITARG,XP,VXP,BV,PBV,TP1,TP2,UM)                     MAI11670
      WRITE OUTPUT TAPE NUTS,810,((PBV(J,I),I=1,6),J=1,3)               MAI11680
  810 FORMAT(40H0 B*T,B*R,VINF PARTIALS AT END POINT PBV                MAI11690
     1/(6E17.8))                                                        MAI11700
      CALL MATRX(PBV,P,PUN,3,6,6,1)                                     MAI11710
      WRITE OUTPUT TAPE NUTS,811,PUN                                    MAI11720
  811 FORMAT(47H0 COVARIANCE MATRIX OF KNOWLEGE OF B*T,B*R,VINF         MAI11730
     1/(3E20.8))                                                        MAI11740
      IGUID = IGUID                                                     MAI11750
      GO TO (23,22),IGUID                                               MAI11760
   22 CONTINUE                                                          MAI11770
      CALL MATRX(AO,PFTA,PFTA,6,6,6,1)                                  MAI11780
      DO 31 I=1,6                                                       MAI11781
      DO 31 J=I,6                                                       MAI11782
      PFTA(I,J)=(PFTA(I,J)+PFTA(J,I))/2.                                MAI11784
      PFTA(J,I)=PFTA(I,J)                                               MAI11786
   31 CONTINUE                                                          MAI11788
      WRITE OUTPUT TAPE NUTS,804,((PTFA(I,J),J=1,6),I=1,6)              MAI11790
  804 FORMAT(38H0 COVARIANCE MATRIX OF GUIDANCE ERRORS                  MAI11800
     1/(6E17.8))                                                        MAI11810
      CALL INVAO(AO,AOI)                                                MAI11820
      CALL MATRX(TRANS,AOI,TRANS,3,6,6,0)                               MAI11830
```

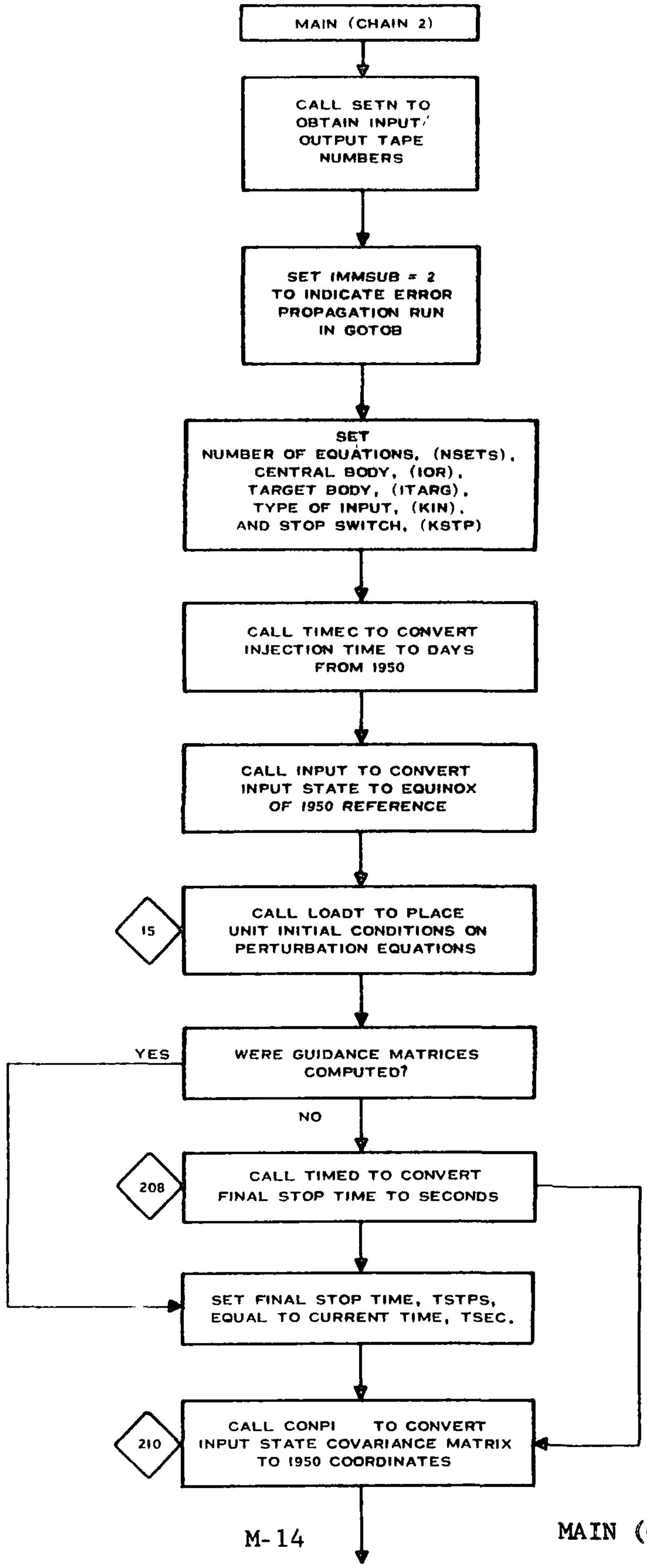
MAIN (CHAIN 1) (Cont'd)

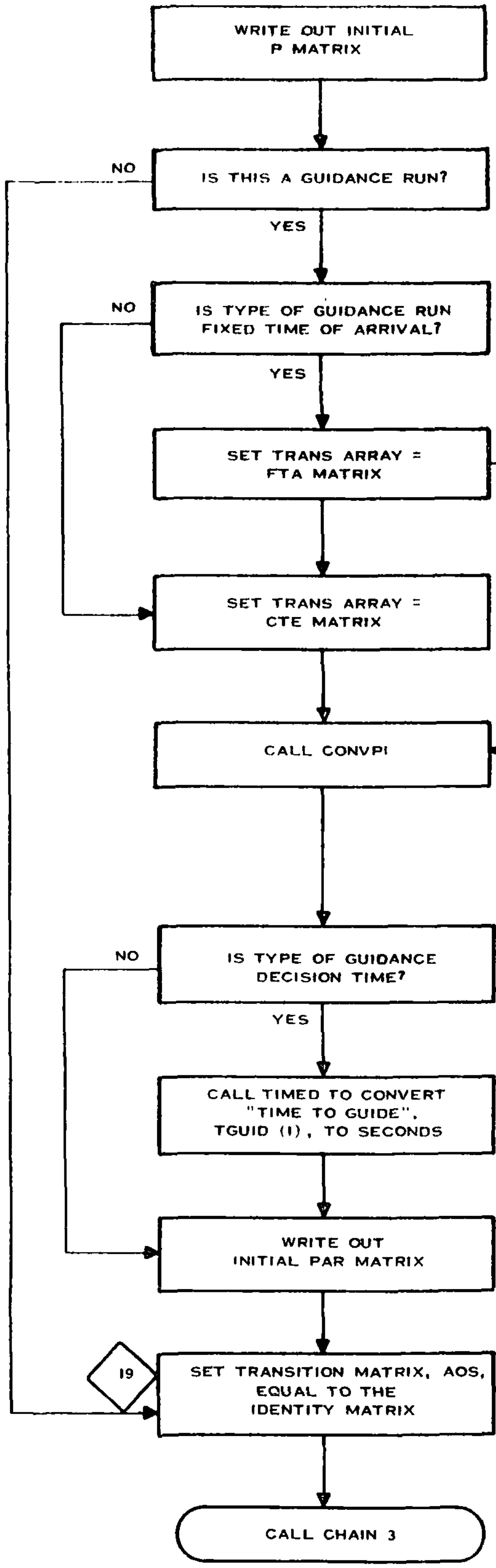
```
      CALL MATRX(TRANS,P,PUN,3,6,6,1)                                   MAI1184
      WRITE OUTPUT TAPE NUTS,803,PUN                                    MAI1185
 803  FORMAT(58H0 COVARIANCE MATRIX OF KNOWLEDGE OF TARGET MISS PARAMETEMAI1186
     1RS                                                                MAI1187
     2/(3E20.8))                                                        MAI1188
      CALL MATRX(TRANS,PFTA,PUN,3,6,6,1)                                MAI1189
      WRITE OUTPUT TAPE NUTS,807,PUN                                    MAI1190
 807  FORMAT(45H0 COVARIANCE MATRIX OF TARGET MISS PARAMETERS           MAI1191
     1/(3E20.8))                                                        MAI1192
      IF(RETR-.0001)23,23,10                                            MAI1192
  10  CONTINUE                                                          MAI1192
      CALL RETRO                                                        MAI1192
  23  CONTINUE                                                          MAI1193
      RETR=0.                                                           MAI1193
      GUID=0                                                            MAI1194
      CIOMP=0                                                           MAI1195
      GO TO 200                                                         MAI1196
      END                                                               MAI1
```

Subroutine: MAIN (Chain 2)

Purpose: Set up program to perform error propagation run. Call Chain 3 to make data run.

Common storages used or required:	T, S, C, IC
Subroutines required:	CHAIN, CONVPI, INPUT, TIMEC, TIMED
Functions required:	None
Approximate number of storages required:	317 DEC


MAIN (CHAIN 2)
CALL SETN TO OBTAIN INPUT/ OUTPUT TAPE NUMBERS
SET IMMSUB = 2 TO INDICATE ERROR PROPAGATION RUN IN GOTOB
SET NUMBER OF EQUATIONS, (NSETS), CENTRAL BODY, (IOR), TARGET BODY, (ITARG), TYPE OF INPUT, (KIN), AND STOP SWITCH, (KSTP)
CALL TIMEC TO CONVERT INJECTION TIME TO DAYS FROM 1950
CALL INPUT TO CONVERT INPUT STATE TO EQUINOX OF 1950 REFERENCE
15
CALL LOADT TO PLACE UNIT INITIAL CONDITIONS ON PERTURBATION EQUATIONS
WERE GUIDANCE MATRICES COMPUTED?
YES
NO
208
CALL TIMED TO CONVERT FINAL STOP TIME TO SECONDS
SET FINAL STOP TIME, TSTPS, EQUAL TO CURRENT TIME, TSEC.
210
CALL CONPI TO CONVERT INPUT STATE COVARIANCE MATRIX TO 1950 COORDINATES


WRITE OUT INITIAL
P MATRIX
IS THIS A GUIDANCE RUN?
NO
YES
IS TYPE OF GUIDANCE RUN
FIXED TIME OF ARRIVAL?
NO
YES
SET TRANS ARRAY =
FTA MATRIX
SET TRANS ARRAY =
CTE MATRIX
CALL CONVPI
IS TYPE OF GUIDANCE
DECISION TIME?
NO
YES
CALL TIMED TO CONVERT
"TIME TO GUIDE",
TGUID (I), TO SECONDS
WRITE OUT
INITIAL PAR MATRIX
19
SET TRANSITION MATRIX, AOS,
EQUAL TO THE
IDENTITY MATRIX
CALL CHAIN 3

MAIN (CHAIN 2)

```
*      LABEL                                                              MAI2
*      SYMBOL TABLE                                                       MAI2
CEC20E2       PART 2 MAIN CONTINUED                                       MAI20000
       COMMON T,S,C,IC                                                    MAI20010
       DIMENSION T(1360),S(1000),C(1000),IC(300),X(3)                     MAI20020
      1,VX(3),XP(3),VXP(3),AN(3,3),UM(6),BV(4),PBV(3,6)                   MAI20030
      2,AO(6,6),PARI(21),PI(21),ISTAT(20),FTA(3,6)                        MAI20040
      3,TRANS(3,6)                                                        MAI20050
      4,PFTA(6,6),P(6,6),TGUID(6)                                         MAI20060
       DIMENSION AOS(6,6),RAD(6), RBOP(6),CTE(3,6),AOI(6,6),PUN(3,3)      MAI20070
       EQUIVALENCE(T,TDUM),(S,SDUM),(C,CDUM),(IC,IDUM),                   MAI20080
      1(IC(8),IMMSUB),(S(112),DATE),(S(113),FDATE),(S(115),X),(S(118),VX)MAI20090
      2,(IC(193),IGUID),(S(110),TSTOP),(IC(2),IOR),(S(111),TIM)           MAI20100
      3,(IC(6),ITARG),(IC(4),KSTP),(IC(1),NSETS),(C(10),TW),(C(11),TF)    MAI20110
      4,(C(138),AN),(C(12),TSECO),(S(71),STD),(S(1),UM),(C(15),XP)        MAI20120
      5,(C(18),VXP),(C(13),TP1),(C(14),TP2),(S(671),PI),(S(650),PARI)     MAI20130
      6,(IC(194),ISTAT),(IC(7),KOUT),(C(649),TSECP)                       MAI20140
      7,(C(460),PFTA),(C(652),P)                                          MAI20150
      8,(S(114),CIBDY),(S(121),TIBDY),(S(722),FTA),(S(124),SKTB)          MAI20160
      9,(S(720),PUPIN),(S(721),PARIN)                                     MAI20170
       EQUIVALENCE (C(752),AOS),(S(900),RAD),(C(112),RBOP),(C(30),TSEC)   MAI20180
      1,(S(740),CTE),(C(648),TGUIDE),(C(850),AOI),(C(850),PUN)            MAI20190
       EQUIVALENCE (S(476),TGUID),(IC(219),IGD)                           MAI20200
      1,(IC(218),IGDKY),(IC(217),IGDTP),(C(568),TRANS)                    MAI20210
       EQUIVALENCE (T(1358),TSTP),(C(797),COMP)                           MAI20220
       CALL SETN(NIN,NOUT)                                                MAI20225
       IMMSUB=2                                                           MAI20230
       NSETS=7                                                            MAI20240
       IOR  =S(114)                                                       MAI20250
       ITARG=S(121)                                                       MAI20260
       KIN  =S(111)                                                       MAI20270
       KSTP=S(124)                                                        MAI20280
       CALL TIMEC(DATE,FDATE,TW,TF)                                       MAI20290
       CALL INPUT(KIN,XP,VXP,TW,TF,AN,X,VX)                               MAI20300
       TSECO =0.                                                          MAI20310
       DO 15  I=1,20                                                      MAI20320
       ISTAT(I)=0                                                         MAI20330
   15  CONTINUE                                                           MAI20340
       KOUT=S(122)                                                        MAI20350
       CALL LOADT                                                         MAI20360
       TSECP=0.                                                           MAI20370
       IF(COMP)208,208,209                                                MAI20380
  208  CONTINUE                                                           MAI20390
       CALL TIMED(TSTOP,TSTPS)                                            MAI20400
       GO TO 210                                                          MAI20410
  209  TSTPS=TSEC                                                         MAI20420
  210  CONTINUE                                                           MAI20430
       TSTP=TSTPS*STD                                                     MAI20440
       INPUP=PUPIN                                                        MAI20450
       CALL CONVPI(INPUP,PI,P)                                            MAI20460
       WRITE OUTPUT TAPE NOUT,708                                         MAI20470
```

AIN (CHAIN 2) (Cont'd)

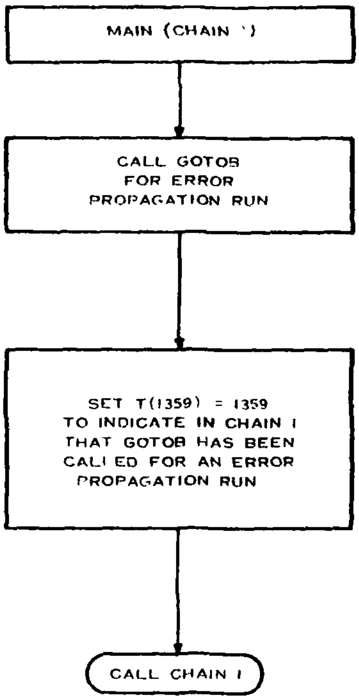
```
708   FORMAT(18H0INITIAL  P MATRIX)                                  MAI20480
      WRITE OUTPUT TAPE NOUT,709,P                                    MAI20490
709   FORMAT(6E17.8)                                                  MAI20500
      IGUID=IGUID                                                     MAI20510
      GO TO (19,18), IGUID                                            MAI20520
  18  CONTINUE                                                        MAI20530
      IGDTP=IGDTP                                                     MAI20540
      GO TO(211,212,212),IGDTP                                        MAI20550
211   CONTINUE                                                        MAI20560
      DO 202 I=1,3                                                    MAI20570
      DO 202 J=1,6                                                    MAI20580
      TRANS(I,J)=FTA(I,J)                                             MAI20590
202   CONTINUE                                                        MAI20600
      GO TO 214                                                       MAI20610
212   DO 213 I=1,3                                                    MAI20620
      DO 213 J=1,6                                                    MAI20630
213   TRANS(I,J)=CTE(I,J)                                             MAI20640
214   CONTINUE                                                        MAI20650
      IGDKY=1                                                         MAI20660
      INPAR=PARIN                                                     MAI20670
      CALL CONVPI(INPAR,PARI,PFTA)                                    MAI20680
      S(475) CONTAINS TYPE OF GUIDANCE DECISION 0 TIME + RATIO        MAI20690
      IF(S(475))17,17,20                                              MAI20700
 17   CONTINUE                                                        MAI20710
      CALL TIMED(TGUID(1),TGUIDE)                                     MAI20720
 20   CONTINUE                                                        MAI20730
      WRITE OUTPUT TAPE NOUT,710                                      MAI20740
710   FORMAT(20H0 INITIAL PAR MATRIX)                                 MAI20750
      WRITE OUTPUT TAPE NOUT,709,PFTA                                 MAI20760
 19   CONTINUE                                                        MAI20770
      DO 150 I=1,6                                                    MAI20780
      DO 151 J=1,6                                                    MAI20790
151   AOS (I,J) =0.                                                   MAI20800
150   AOS (I,I)=1.                                                    MAI20810
      CALL CHAIN(3,B3)                                                MAI20820
      END                                                             MAI2
```

Page Intentionally Left Blank

Subroutine: MAIN (Chain 3)

Purpose: Call subroutine GOTOB to perform integration of trajectory for an error propagation run. Set cell T(1359) = 1359 to indicate GOTOB has been called and call Chain 1.

Common storages used or required:	T
Subroutines required:	GOTOB
Functions required:	None
Approximate number of storages required:	


MAIN (CHAIN)
CALL GOTOB
FOR ERROR
PROPAGATION RUN
SET T(1359) = 1359
TO INDICATE IN CHAIN 1
THAT GOTOB HAS BEEN
CALLED FOR AN ERROR
PROPAGATION RUN
CALL CHAIN 1

IN (CHAIN 3)

```
      LABEL                                                   MAI3
      SYMBOL TABLE                                            MAI3
C20C2         CHAIN 3 GOTOB ONLY                              MAI30000
      COMMON T                                                MAI30010
      DIMENSION T(1360)                                       MAI30020
      EQUIVALENCE(T(1358),TSTP)                               MAI30030
      CALL GOTOB(TSTP)                                        MAI30040
      T(1359)=1359.                                           MAI30050
      CALL CHAIN(1,B3)                                        MAI30060
      END                                                     MAI3
```

Page Intentionally Left Blank

SUBROUTINES

(Subroutine discussion is presented in the following manner: (1) subroutine description, (2) subroutine flow diagrams, if applicable, (3) subroutine listings.)

Page Intentionally Left Blank

Function: ARKTAN

Purpose: To find the double precision arctangent θ of (Y/X) in the proper quadrant. $-\Pi \leq \theta \leq \Pi$ for N = 180; $0 \leq \theta \leq 360$ for N = 360.

Calling Sequence:

Z = ARKTAN (N, X, Y)

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	N	1			N = 180 or 360, as above
I	X	1			
I	Y	1			
0	ARKTAN			radians	$\tan^{-1}$ (Y/X)

Common storages used or required: None

Subroutines required: None

Functions required: DATAN, other double precision routines

Approximate number of storages required:

```
*       LABEL                                                                    ARKT
*       SYMBOL TABLE                                                             ARKT
CEC2021  FUNCTION ARKTAN                                                         ARKT0000
      CALL SETN(NIN,NOUT)                                                        ARKT0015
      FUNCTION ARKTAN(N,X,Y)                                                     ARKT0010
D     P=3.14159265                                                               ARKT0020
      K=N/180                                                                    ARKT0030
D     ARKTAN = ATANF(ABSF(Y/X))                                                  ARKT0040
      IF(Y)1,2,3                                                                 ARKT0050
    1 GO TO (4,5),K                                                              ARKT0060
    4 IF (X)10,11,12                                                             ARKT0070
D  10 ARKTAN = ARKTAN-P                                                          ARKT0080
      GO TO 9                                                                    ARKT0090
D  11 ARKTAN = -P*.5                                                             ARKT0100
      GO TO 9                                                                    ARKT0110
D  12 ARKTAN = -ARKTAN                                                           ARKT0120
      GO TO 9                                                                    ARKT0130
    5 IF (X)13,14,15                                                             ARKT0140
D  13 ARKTAN = P+ARKTAN                                                          ARKT0150
      GO TO 9                                                                    ARKT0160
D  14 ARKTAN = 3.*.5*P                                                           ARKT0170
      GO TO 9                                                                    ARKT0180
D  15 ARKTAN = 2.*P-ARKTAN                                                       ARKT0190
      GO TO 9                                                                    ARKT0200
    2 IF (X)16,17,9                                                              ARKT0210
D  16 ARKTAN = P                                                                 ARKT0220
      GO TO 9                                                                    ARKT0230
   17 WRITE OUTPUT TAPE NOUT,18                                                  ARKT0240
   18 FORMAT (90H0                                   THE ARKTAN FUNCTION HAS     ARKT0250
     1FAILED  -  ARKTAN(0/0) IS UNDEFINED/1H0)                                   ARKT0260
      GO TO 9                                                                    ARKT0270
    3  IF(X)8,19,9                                                               ARKT0280
D    8 ARKTAN = P-ARKTAN                                                         ARKT0290
      GO TO 9                                                                    ARKT0300
D 19  ARKTAN = .5*P                                                              ARKT0310
    9 RETURN                                                                     ARKT0320
      END                                                                        ARKT
```

Function: ARKTNS

Purpose: To find the arctangent θ of (Y/X) in the proper quadrant. $-\Pi \le \theta \le \Pi$ for N = 180; $0 \le \theta < 360$ for N = 360.

Calling Sequence:

Z = ARKTNS (N, X, Y)

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	N	1			N = 180 or 360, as above
I	X	1			
I	Y	1			
O	ARKTNS			radians	$\tan^{-1}$ (Y/X)

Common storages used or required: None

Subroutines required: None

Functions required: ATANF

Approximate number of storages required:

```
*      LABEL                                                          ARKS
*      SYMBOL TABLE                                                   ARKS
CEC2022  FUNCTION ARKTNS                                              ARKS0000
       FUNCTION ARKTNS(N,X,Y)                                         ARKS0010
       CALL SETN(NIN,NOUT)                                            ARKS0015
       P=3.14159265                                                   ARKS0020
       K=N/180                                                        ARKS0030
       ARKTNS = ATANF(ABSF(Y/X))                                      ARKS0040
       IF(Y)1,2,3                                                     ARKS0050
     1 GO TO (4,5),K                                                  ARKS0060
     4 IF (X)10,11,12                                                 ARKS0070
    10 ARKTNS = ARKTNS-P                                              ARKS0080
       GO TO 9                                                        ARKS0090
    11 ARKTNS = -P*.5                                                 ARKS0100
       GO TO 9                                                        ARKS0110
    12 ARKTNS = -ARKTNS                                               ARKS0120
       GO TO 9                                                        ARKS0130
     5 IF (X)13,14,15                                                 ARKS0140
    13 ARKTNS = P+ARKTNS                                              ARKS0150
       GO TO 9                                                        ARKS0160
    14 ARKTNS = 3.*.5*P                                               ARKS0170
       GO TO 9                                                        ARKS0180
    15 ARKTNS = 2.*P-ARKTNS                                           ARKS0190
       GO TO 9                                                        ARKS0200
     2 IF (X)16,17,9                                                  ARKS0210
    16 ARKTNS = P                                                     ARKS0220
       GO TO 9                                                        ARKS0230
    17 WRITE OUTPUT TAPE NOUT,18                                      ARKS0240
    18 FORMAT (90H0                       THE ARKTAN FUNCTION HAS     ARKS0250
      1FAILED  -  ARKTAN(0/0) IS UNDEFINED/1H0)                       ARKS0260
       GO TO 9                                                        ARKS0270
    3  IF(X)8,19,9                                                    ARKS0280
     8 ARKTNS = P-ARKTNS                                              ARKS0290
       GO TO 9                                                        ARKS0300
   19  ARKTNS = .5*P                                                  ARKS0310
     9 RETURN                                                         ARKS0320
       END                                                            ARKS
```

Function: ASINH

Purpose: To compute θ = arcsinh (X) = $\log_e (X + \sqrt{X^2 + 1})$

Calling Sequence:

Y = ASINH (X)

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	X				
O	ASINH			radians	$\sinh^{-1}(X) = \log_e (X + \sqrt{X^2 + 1})$

Common storages used or required: None

Subroutines required: None

Functions required: LOGF

Approximate number of storages required:

```
*      LABEL                                              ASIN
*      SYMBOL TABLE                                       ASIN
       FUNCTION ASINH(X)                                  ASIN000
       ASINH=LOGF(X+SQRTF(X*X+1.0))                       ASIN001
       RETURN                                             ASIN002
       END                                                ASIN
```

Subroutine: BODY

Purpose: To compute the perturbing acceleration due to the perturbing bodies included in the ephemeris. It also computes the first variation of the perturbing acceleration for use in the integration of the variational equations.

Calling Sequence:

CALL BODY (XP, PO, VKB, RBO, RBOP, CA, NOR, BP, T, NEQ, U)

Input and **Output**

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	XP	3	$\vec{R}$	Km	Position vector
I	PO	22	$\vec{R}_i$	Km	Positions of bodies in ephemeris
I	VKB	6	μ_i	Km^3/sec^2	Gravitational constants of bodies
					being considered
0	RBOP	6	$\lvert\vec{R}\rvert_{iv}$	Km	Magnitude of position of vehicle
					relative to i th body
0	RBO	6	$\lvert\vec{R}\rvert_i$	Km	Position of $i^{\underline{th}}$ body relative to
					central body

(Cont'd.)

Common storages used or required: None

Subroutines required: None

Functions required: FNORM

Approximate number of storages required: 445 DEC

(Cont'd.)

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
O	CA	3			Perturbation accelerations
I	NOR	1			Number of central body
O	BP	3,3	$\frac{\partial \ddot{x}_i}{\partial x_j}$		First variation of perturbation
					accelerations
I/O	T	1360			Common T Block
I	NEQ	1			Number of equations being
					integrated
I	U	1	μ	Km^3/sec^2	Central body gravitational
					constant

Equation Being Solved

The perturbation term due to the n bodies being considered is:

$$\vec{P} = -\sum_{j=1}^{n} \mu_j \quad \frac{\vec{R}_{jp}}{R_{jp}^3} + \frac{\vec{R}_j}{R_j^3}$$

where $\vec{R}_{jp} = \vec{R} - \vec{R}_j$.

The vector $\vec{P}$ is placed in the cells called CA.

If the number of equations being solved is greater than three, the first variation of $\vec{P}$ with respect to position is determined and placed in the T block for the integration package

$$AP = \frac{\partial \ddot{X}_i}{\partial X_j} \qquad \begin{array}{l} i = 1,3 \\ j = 1,3 \end{array}$$

The form of AP is obtained by differentiating

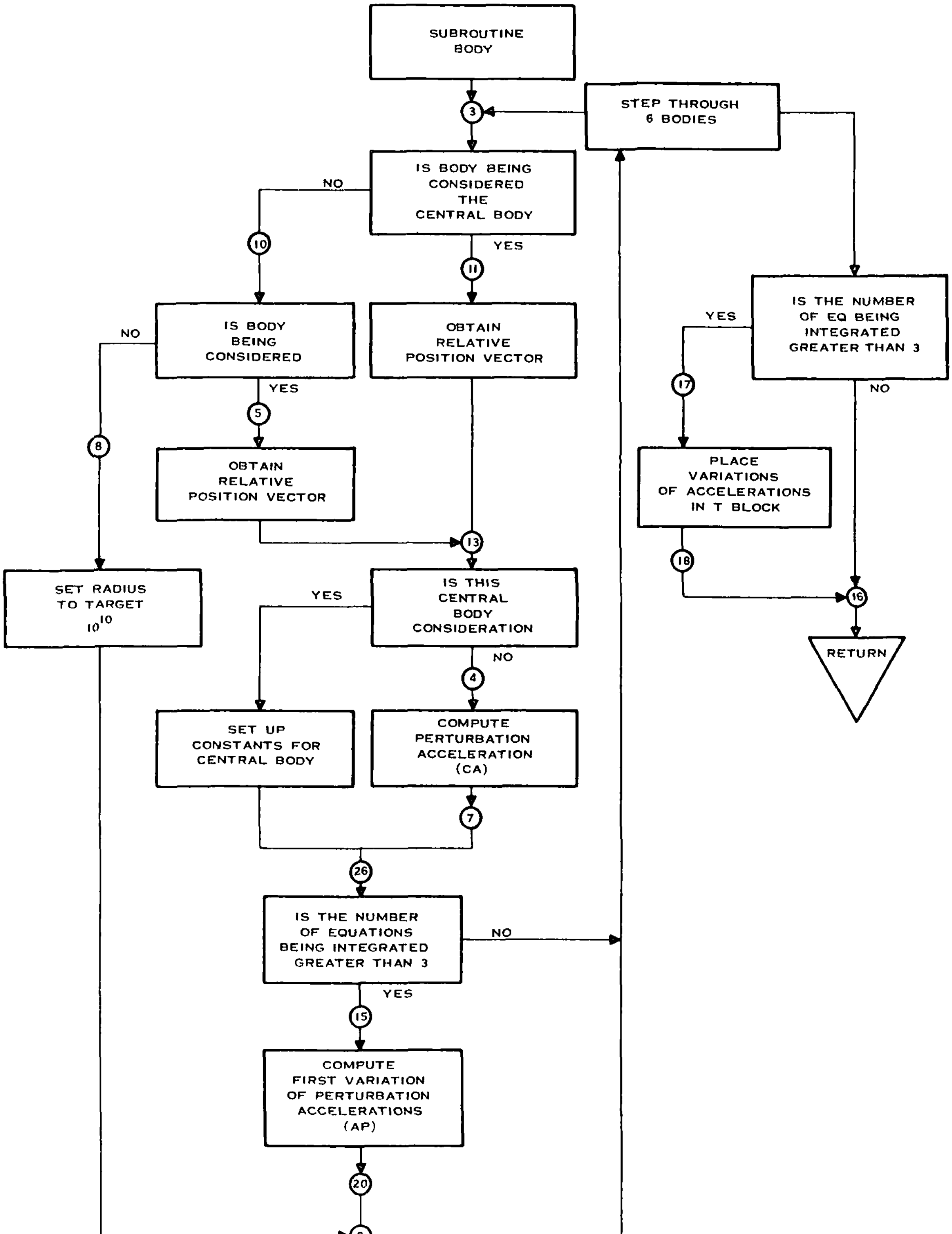
$$-\mu \frac{\vec{R}}{R^3} + \vec{P}$$

with respect to X_j. This yields:

$$AP_{ij} = -\sum_{k=0}^{n} \mu_k \left\{ \frac{1}{R_{kp}^3} \frac{\partial \vec{R}}{\partial X_j} - \frac{3}{R_{kp}^5} \left(\vec{R}_{kp} \cdot \frac{\partial \vec{R}}{\partial X_j} \right) \vec{R}_{kp} \right\}$$

with $\mu_o = \mu$ and $\vec{R}_{op} = \vec{R}$

$$\text{or } AP_{ij} = \sum_{k=0}^{n} \frac{\mu_k}{R_{kp}^3} \left\{ 3 \; \underline{\vec{R}_{kp} \, \vec{R}_{kp}^T} - I \right\}$$


SUBROUTINE BODY
3
STEP THROUGH 6 BODIES
IS BODY BEING CONSIDERED THE CENTRAL BODY
NO
YES
10
11
IS BODY BEING CONSIDERED
OBTAIN RELATIVE POSITION VECTOR
NO
YES
5
8
OBTAIN RELATIVE POSITION VECTOR
13
SET RADIUS TO TARGET 10^{10}
IS THIS CENTRAL BODY CONSIDERATION
YES
NO
4
SET UP CONSTANTS FOR CENTRAL BODY
COMPUTE PERTURBATION ACCELERATION (CA)
7
26
IS THE NUMBER OF EQUATIONS BEING INTEGRATED GREATER THAN 3
NO
YES
15
COMPUTE FIRST VARIATION OF PERTURBATION ACCELERATIONS (AP)
20
9
IS THE NUMBER OF EQ BEING INTEGRATED GREATER THAN 3
YES
NO
17
PLACE VARIATIONS OF ACCELERATIONS IN T BLOCK
18
16
RETURN

```
*     LABEL
*     SYMBOL TABLE
      SUBROUTINE BODY(X,PO,VKB,RBO,RBOP,CA,NOR,AP,T,NEQ,U)
      DIMENSION X(3),PO(22),VKB(6),RBO(6),RBOP(6),CA(3),XN(18),RJP(3),
     1AP(3,3),T(1360)
      N = NOR
      DO 1 I=1,15
      J = 23-I
    1 XN(I) = PO(J)
      DO 2 I=16,18
      J = 20-I
    2 XN(I) = PO(J)
      DO 3 I=1,3
      CA(I)=0.
    3 CONTINUE
      DO 9 I=1,6
      K1 = 3*I-3
      IF(I-N)10,11,10
   11 CONTINUE
      DO 12 J=1,3
      RJP(J)=X(J)
   12 CONTINUE
      GO TO 13
   10 CONTINUE
      IF (VKB(I)) 5,8,5
    5 DO 6 J=1,3
      K = K1+J
      RJP(J) = X(J) - XN(K)
    6 CONTINUE
   13 CONTINUE
      RBOP(I) = FNORM(RJP)
      RBO(I) = FNORM(XN(K1+1))
      R3 = RBOP(I)**3
      X3 = RBO(I)**3
      FAC=VKB(I)/R3
      FAX=VKB(I)/X3
      IF(I-N)4,23,4
    4 CONTINUE
C     ACCELERATIONS (NEGATIVE) DUE TO PERTURBING BODIES
      DO 7 J=1,3
      K1J = K1 + J
      CA(J)=CA(J)+FAC*RJP(J)+FAX*XN(K1J)
    7 CONTINUE
      GO TO 26
   23 CONTINUE
      FAC=U/R3
      FAX=U/X3
   26 CONTINUE
      DO 24 J=1,3
      RJP(J)=RJP(J)/RBOP(I)
   24 CONTINUE
```

```
C       RJP IS NOW A UNIT VECTOR
        IF(NEQ-3)14,14,15
   15   CONTINUE
C       PARTIALS FOR PERTURBATION EQUATIONS
        DO 20 J=1,3
        J=J
        DO 20 K=J,3
        AP(J,K)=AP(J,K)+FAC*(3.*RJP(J)*RJP(K))
        IF(J-K)21,22,21
   21   CONTINUE
        AP(K,J)=AP(J,K)
        GO TO 20
   22   CONTINUE
        AP(J,J)=AP(J,J)-FAC
   20   CONTINUE
   25   FORMAT(3E16.8)
   14   CONTINUE
        GO TO 9
    8   RBOP(I) = 1.E10
    9   CONTINUE
        NS=NEQ/3-1
        IF(NS)16,16,17
   16   CONTINUE
        RETURN
   17   CONTINUE
        DO 18 I=1,NS
        NST=2*NEQ+3+3*I
        NSTP=3+3*I
        DO 19 J=1,3
        KK=NST+J
        T(KK)=0.
        DO 19 K=1,3
        JJ=NSTP+K
        T(KK)=T(KK)+AP(J,K)*T(JJ)
   19   CONTINUE
   18   CONTINUE
        GO TO 16
        END
```

Subroutine: BVEC

Purpose: To obtain the matrix of target miss partials; B·T, B·R, and V infinity with respect to the state.

Calling Sequence:

CALL BVEC (IOR, ITARG, X, V, BV, PBV, TW, TF, UM)

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	IOR	1			Central body
I	ITARG	1			Target body
I	X	3	$\vec{R}$		Position vector
I	V	3	$\vec{\upsilon}$		Velocity vector
0	BV	4			Nominal Miss Parameters
0	PBV	(3,6)			Matrix of Miss Partials
I	TW	1	t	days	Whole days from 1950
I	TF	1	t	days	Fractional days from 1950
I	UM	1	μ	Km^3/sec^2	Gravitational constant

Common storages used or required: None

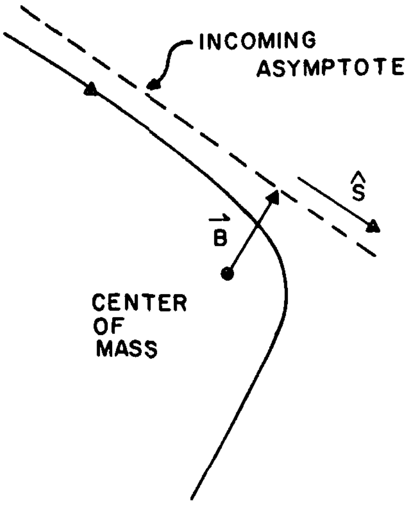
Subroutines required: DOT, INTRI, CROSS

Functions required: FNORM

Approximate number of storages required:

Derivation of the $\vec{B}$, $\vec{S}$, $\vec{T}$, and $\vec{R}$ vectors

The B, R, S, and T vectors are used to define a target miss (see Fig. 1 below)



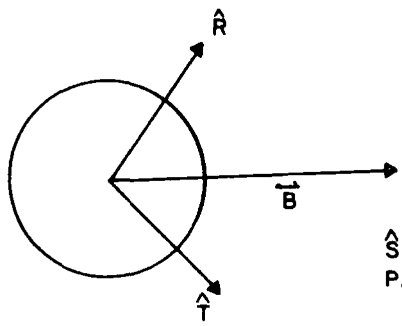


FIG. I

$\vec{B}$, the position vector in the plane of the trajectory originating at the center of gravity of the target and directed perpendicularly to the incoming asymptote of the hyperbola, is approximately the vector miss which would occur if the target had no mass.

To determine the target miss direction relative to the target, a set of orthogonal vectors is computed. Let $\hat{S}$ be a unit vector in the direction of the incoming asymptote, $\hat{T}$ be a unit vector perpendicular to $\hat{S}$ that lies in a fixed plane of interest such as the equatorial plane, and $\hat{R}$ be a unit vector to form a right-handed system.

$$\hat{R} = \hat{S} \times \hat{T}$$

Since $\vec{B}$ is perpendicular to $\hat{S}$, it lies in the plane determined by $\hat{R}$ and $\hat{T}$. The variables which are used to describe the miss relative to a plane of interest are the projections of $\vec{B}$ on $\hat{T}$ and $\hat{R}$, i.e., $\vec{B}\cdot\hat{T}$ and $\vec{B}\cdot\hat{R}$.

The quantities used in the derivation of the vectors are the following:

$$h = |\vec{R} \times \vec{v}| \qquad C_3 = v^2 - \frac{2\mu}{R}$$

$$e = \sqrt{1 + \frac{h^2 \, C3}{\mu^2}} \qquad P = \frac{h^2}{\mu}$$

True anamoly

$$\theta = \cos^{-1}\left[\frac{1}{\epsilon}\left(\frac{P}{R} - 1\right)\right]$$

Unit vectors to specify orientation of conic

$$\zeta = \frac{\vec{R} \times \vec{v}}{|\vec{R} \times \vec{v}|} \qquad \vec{M} = \vec{\zeta} \times \frac{\vec{R}}{|\vec{R}|}$$

$$\vec{\xi} = \frac{\vec{R}}{|\vec{R}|} \cos\theta - \vec{M} \sin\theta$$

$$\vec{h} = \vec{\zeta} \times \vec{\xi}$$

The unit vector in the direction of the asymptote is given by

$$\hat{S} = \frac{1}{e}\,\vec{\xi} + \sqrt{1 - \left(\frac{1}{e}\right)^2}\;\vec{n}$$

The impact parameter

$$\vec{B} = |\vec{B}| \left[\sqrt{1 - (\frac{1}{e})^2}\ \vec{\xi} - \frac{1}{e}\ \vec{n} \right]$$

where $|\vec{B}| = \frac{\mu}{C3} \sqrt{e^2 - 1}$

Also

$$T_X = \frac{S_Y}{\sqrt{S_X^{\ 2} + S_Y^{\ 2}}}$$

$$T_Y = - \frac{S_X}{\sqrt{S_X^{\ 2} + S_Y^{\ 2}}}$$

$$T_Z = 0$$

and

$$\hat{R} = \hat{S} \times \hat{T}$$

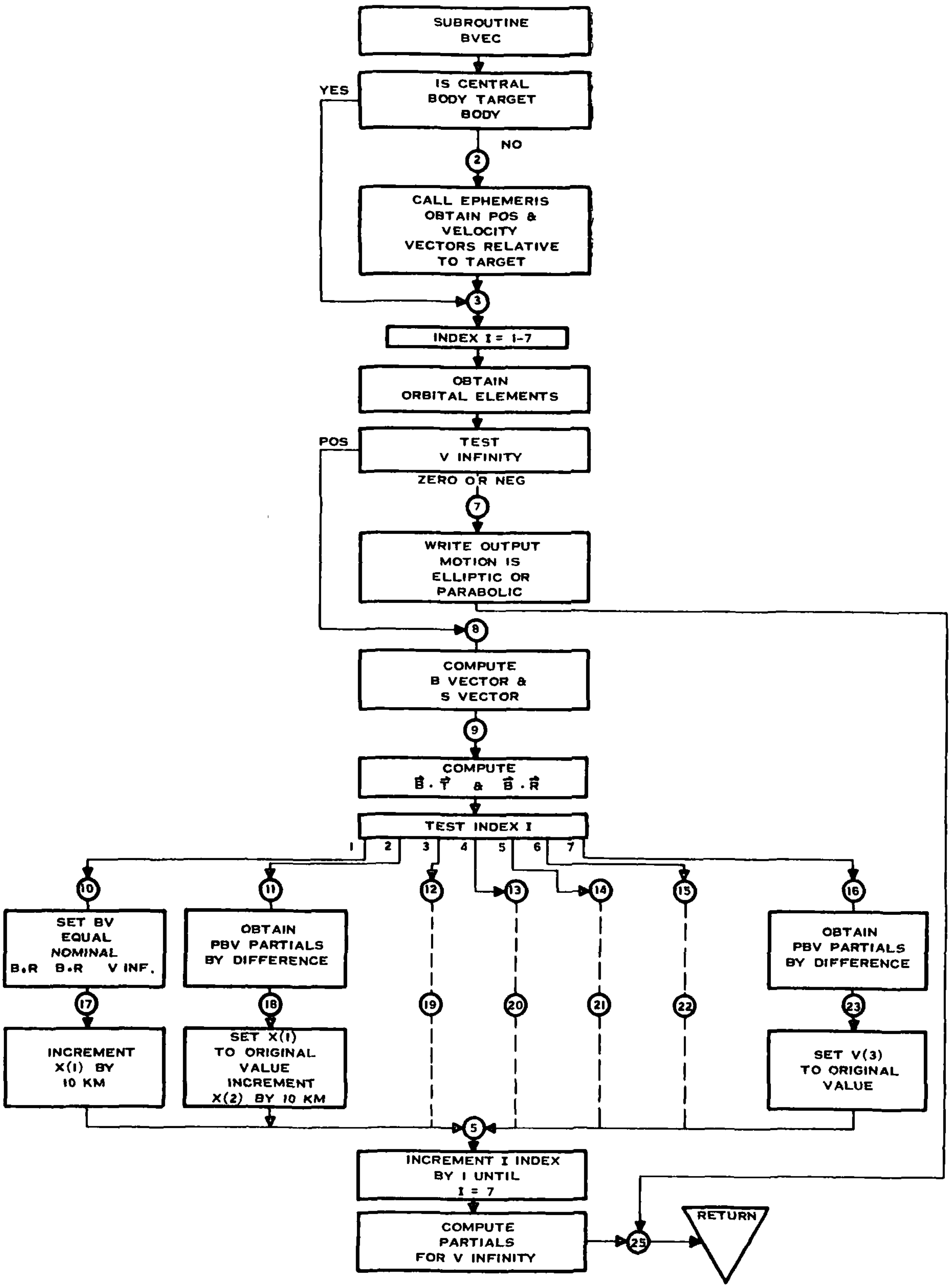
Derivation of partials of V infinity with respect to the state

$$VINF = \sqrt{v^2 - \frac{2\mu}{R}}$$

$$\frac{\partial VINF}{\partial X_i} = \frac{\mu\ X_i}{(VINF)\ R^3} \qquad i = 1,2,3$$

$$\frac{\partial VINF}{\partial X_i} = \frac{X_i}{(VINF)} \qquad i = 1,2,3$$

Reference: JPL EXTERNAL PUBLICATION NO. 674


SUBROUTINE BVEC
IS CENTRAL BODY TARGET BODY
YES
NO
2
CALL EPHEMERIS OBTAIN POS & VELOCITY VECTORS RELATIVE TO TARGET
3
INDEX I = 1-7
OBTAIN ORBITAL ELEMENTS
TEST V INFINITY
POS
ZERO OR NEG
7
WRITE OUTPUT MOTION IS ELLIPTIC OR PARABOLIC
8
COMPUTE B VECTOR & S VECTOR
9
COMPUTE $\vec{B} \cdot \hat{T}$ & $\vec{B} \cdot \hat{R}$
TEST INDEX I
1 2 3 4 5 6 7
10
SET BV EQUAL NOMINAL B.R B.R V INF.
11
OBTAIN PBV PARTIALS BY DIFFERENCE
12
13
14
15
16
OBTAIN PBV PARTIALS BY DIFFERENCE
17
INCREMENT X(I) BY 10 KM
18
SET X(I) TO ORIGINAL VALUE INCREMENT X(2) BY 10 KM
19
20
21
22
23
SET V(3) TO ORIGINAL VALUE
5
INCREMENT I INDEX BY I UNTIL I = 7
COMPUTE PARTIALS FOR V INFINITY
25
RETURN

```
*       LABEL                                                                BVEC
*       SYMBOL TABLE                                                         BVEC
CEC2001  SUBROUTINE BVEC                                                     BVEC0000
      SUBROUTINE BVEC (IOR,ITARG,X,V,BV,PBV,TW,TF,UM)                        BVEC0010
      DIMENSION X(3),V(3),XP(3),VP(3),BV(4),PBV(3,6),PO(22),VE(22)           BVEC0020
     1,BD(3),CP(3),RB(3),RM(3),UM(6),PP(3),QQ(3),S(3),B(3),RR(3),TT(3)       BVEC0030
      CALL SETN(NIN,NOUT)                                                    BVEC0035
      D=2.E6                                                                 BVEC0040
      DO 1 I=1,3                                                             BVEC0050
      XP(I)=X(I)                                                             BVEC0060
      VP(I)=V(I)                                                             BVEC0070
    1 CONTINUE                                                               BVEC0080
      IF(IOR-ITARG) 2,3,2                                                    BVEC0090
    2 CONTINUE                                                               BVEC0100
      N=IOR-1                                                                BVEC0110
      CALL INTR1(TW,TF,N,PO,1,VE,D)                                          BVEC0120
      IC=26-3*ITARG                                                          BVEC0130
      DO 4 I=1,3                                                             BVEC0140
      IB=IC-I                                                                BVEC0150
      XP(I)=XP(I)-PO(IB)                                                     BVEC0160
      VP(I)=VP(I)-VE(IB)                                                     BVEC0170
    4 CONTINUE                                                               BVEC0180
    3 CONTINUE                                                               BVEC0190
      DO 5 I=1,7                                                             BVEC0200
      CALL CROSS(XP,VP,CP)                                                   BVEC0210
      D=FNORM(CP)                                                            BVEC0220
      R=FNORM(XP)                                                            BVEC0230
      DO 6 J=1,3                                                             BVEC0240
      CP(J)=CP(J)/D                                                          BVEC0250
      RB(J)=XP(J)/R                                                          BVEC0260
    6 CONTINUE                                                               BVEC0270
      CALL CROSS(CP,RB,RM)                                                   BVEC0280
      U=UM(ITARG)                                                            BVEC0290
      C3=DOT(VP,VP)-2.*U/R                                                   BVEC0300
      IF(C3)7,7,8                                                            BVEC0310
    7 CONTINUE                                                               BVEC0320
      WRITE OUTPUT TAPE NOUT,100                                             BVEC0330
  100 FORMAT(32H MOTION IS ELLIPTIC OR PARABOLIC)                            BVEC0340
      GO TO 25                                                               BVEC0350
    8 CONTINUE                                                               BVEC0360
      P=D**2/U                                                               BVEC0370
      E2=1.+P*C3/U                                                           BVEC0380
      E = SQRTF(E2)                                                          BVEC0390
      A=U/C3                                                                 BVEC0400
      SEM2=E2-1.                                                             BVEC0410
      SEM1=SQRTF(SEM2)                                                       BVEC0420
      ASE=A*SEM2/E                                                           BVEC0430
      SE =SEM1/E                                                             BVEC0440
      RDOT=DOT(XP,VP)/R                                                      BVEC0450
      CTHET=(P-R)/(E*R)                                                      BVEC0460
      STHET=(RDOT*SQRTF(P/U))/E                                              BVEC0470
```

BVEC - 6

```
      DO 9 J=1,3                                                        BVEC0480
      PP(J)=CTHET*RB(J)-STHET*RM(J)                                     BVEC0490
      QQ(J)=STHET*RB(J)+CTHET*RM(J)                                     BVEC0500
      S(J)=PP(J)/E+SE*QQ(J)                                             BVEC0510
      B(J)=ASE*PP(J)-A*SE*QQ(J)                                         BVEC0520
    9 CONTINUE                                                          BVEC0530
      S2=SQRTF(S(1)**2+S(2)**2)                                         BVEC0540
      TT(1)=S(2)/S2                                                     BVEC0550
      TT(2)=-S(1)/S2                                                    BVEC0560
      TT(3)=0.                                                          BVEC0570
      CALL CROSS(S,TT,RR)                                               BVEC0580
      BD(1)=DOT(B,TT)                                                   BVEC0590
      BD(2)=DOT(B,RR)                                                   BVEC0600
      I=I                                                               BVEC0610
      GO TO (10,11,12,13,14,15,16),I                                    BVEC0620
   10 CONTINUE                                                          BVEC0630
      DO 17 J=1,2                                                       BVEC0640
      BV(J)=BD(J)                                                       BVEC0650
   17 CONTINUE                                                          BVEC0660
      BV(3)=SQRTF(C3)                                                   BVEC0670
      XP(1)=XP(1)+10.                                                   BVEC0680
      GO TO 5                                                           BVEC0690
   11 CONTINUE                                                          BVEC0700
      DO 18 J=1,2                                                       BVEC0710
      PBV(J,1)=(BD(J)-BV(J))/10.                                        BVEC0720
   18 CONTINUE                                                          BVEC0730
      XP(1)=XP(1)-10.                                                   BVEC0740
      XP(2)=XP(2)+10.                                                   BVEC0750
      GO TO 5                                                           BVEC0760
   12 CONTINUE                                                          BVEC0770
      DO 19 J=1,2                                                       BVEC0780
      PBV(J,2)=(BD(J)-BV(J))/10.                                        BVEC0790
   19 CONTINUE                                                          BVEC0800
      XP(2)=XP(2)-10.                                                   BVEC0810
      XP(3)=XP(3)+10.                                                   BVEC0820
      GO TO 5                                                           BVEC0830
   13 CONTINUE                                                          BVEC0840
      DO 20 J=1,2                                                       BVEC0850
      PBV(J,3)=(BD(J)-BV(J))/10.                                        BVEC0860
   20 CONTINUE                                                          BVEC0870
      XP(3)=XP(3)-10.                                                   BVEC0880
      VP(1)=VP(1)+.01                                                   BVEC0890
      GO TO 5                                                           BVEC0900
   14 CONTINUE                                                          BVEC0910
      DO 21 J=1,2                                                       BVEC0920
      PBV(J,4)=(BD(J)-BV(J))*100.                                       BVEC0930
   21 CONTINUE                                                          BVEC0940
      VP(1)=VP(1)-.01                                                   BVEC0950
      VP(2)=VP(2)+.01                                                   BVEC0960
      GO TO 5                                                           BVEC0970
   15 CONTINUE                                                          BVEC0980
```

```
      DO 22 J=1,2                                                       BVEC0990
      PBV(J,5)=(BD(J)-BV(J))*100.                                       BVEC1000
   22 CONTINUE                                                          BVEC1010
      VP(2)=VP(2)-.01                                                   BVEC1020
      VP(3)=VP(3)+.01                                                   BVEC1030
      GO TO 5                                                           BVEC1040
   16 CONTINUE                                                          BVEC1050
      DO 23 J=1,2                                                       BVEC1060
      PBV(J,6)=(BD(J)-BV(J))*100.                                       BVEC1070
   23 CONTINUE                                                          BVEC1080
      VP(3)=VP(3)-.01                                                   BVEC1090
    5 CONTINUE                                                          BVEC1100
      VIN=SQRTF(C3)                                                     BVEC1110
      UR2=1.*U/R**3                                                     BVEC1120
      UR3=UR2/VIN                                                       BVEC1130
      DO 24 I=1,3                                                       BVEC1140
      J=I+3                                                             BVEC1150
      PBV(3,J)=VP(I)/VIN                                                BVEC1160
      PBV(3,I)=UR3*XP(I)                                                BVEC1170
   24 CONTINUE                                                          BVEC1180
   25 CONTINUE                                                          BVEC1190
      RETURN                                                            BVEC1200
      END                                                               BVEC
```

Subroutine: CHNGP

Purpose: To choose the appropriate body center as a function of distance from the present central body.

Calling Sequence:

CALL CHNGP (NOR, IOR, RAD)

Input and **Output**

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
O	NOR	1			New body center
I	IOR	1			Present body center
I	RAD	6		Km.	RAD(I) = distance from body I
					to vehicle

Common storages used or required: None

Subroutines required: None

Functions required: None

Approximate number of storages required:

Method of Selecting Body Center

Present Central Body	New Central Body	
1) Earth	Sun	$R_1 \geq 25 \times 10^5$
	Earth	$R_2 \geq 3 \times 10^4$
	Moon	$R_2 \leq 3 \times 10^4$
2) Moon	Moon	$R_2 < 3 \times 10^4$
	Earth	$R_2 \geq 3 \times 10^4$
3) Sun	Earth	$R_1 \leq 25 \times 10^5$
	Venus	$R_4 < 25 \times 10^5$
	Mars	$R_5 < 25 \times 10^5$
	Jupiter	$R_6 < 25 \times 10^5$
	Sun	(if none of above)
4) Venus	Venus	$R_4 < 25 \times 10^5$
	Sun	$R_4 \geq 25 \times 10^5$
5) Mars	Mars	$R_5 < 25 \times 10^5$
	Sun	$R_5 \geq 25 \times 10^5$
6) Jupiter	Jupiter	$R_6 \geq 25 \times 10^5$
	Sun	$R_6 < 25 \times 10^5$

where R_i = RAD(I) = distance from body(i) to the vehicle.

```
    LABEL                                                    CHNG
    SYMBOL TABLE                                             CHNG
C2002                                                        CHNG
    SUBROUTINE CHNGP(NOR,IOR,RAD)                            CHNG0000
    DIMENSION RAD(6)                                         CHNG0010
    IO = IOR                                                 CHNG0020
    NOR = IOR                                                CHNG0025
    GO TO (1,2,3,4,4,4), IO                                  CHNG0030
  1 IF (RAD(1)-2500000.) 11,12,12                            CHNG0040
 11 IF (RAD(2)-30000.) 13,14,14                              CHNG0050
 12 NOR = 3                                                  CHNG0060
    GO TO 14                                                 CHNG0070
 13 NOR = 2                                                  CHNG0080
 14 RETURN                                                   CHNG0090
  2 IF (RAD(2)-30000.) 14,21,21                              CHNG0100
 21 NOR = 1                                                  CHNG0110
    GO TO 14                                                 CHNG0120
  3 IF (RAD(1)-2500000.) 31,31,32                            CHNG0130
 31 NOR = 1                                                  CHNG0140
    GO TO 14                                                 CHNG0150
 32 DO 34 I=4,6                                              CHNG0160
    IF (RAD(I)-2500000.) 33,34,34                            CHNG0170
 33 NOR = I                                                  CHNG0180
    GO TO 14                                                 CHNG0190
 34 CONTINUE                                                 CHNG0200
    GO TO 14                                                 CHNG0210
  4 IF (RAD(IO)-2500000.) 14,41,41                           CHNG0220
 41 NOR = 3                                                  CHNG0230
    GO TO 14                                                 CHNG0240
    END                                                      CHNG
```

Page Intentionally Left Blank

Subroutine: COMPHQ

Purpose: To compute the H matrix of measurement partials with respect to the vehicle state and the variance of the measurements. The subroutine is called by ONBTR, on-board tracking subroutine, and MONBTR, the moon beacon tracking subroutine. The types of measurements which are evaluated are range, right ascension, declination, and range rate. After computation of the H matrix and the measurement error, subroutine CORRTP is called to perform the filtering of the measurement and updating of the state covariance matrix, P.

Calling Sequence:

CALL COMPHQ (JJJ, KK, NN)

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	JJJ	1			Type of measurement
I	KK	1			On-board or moon beacon
I	NN	1			Key for locating constants in
					storage

Common storages used or required: T, S, C, IC

Subroutines required: CORRTP, DOT

Functions required: SQRT

Approximate number of storages required: 388 DEC

Calling Sequence Elaboration

The calling sequence of COMPHQ consists of CALL COMPHQ (JJJ, KK, NN) where:

JJJ is a key which indicates the type of measurement being performed.

1. Range
2. Right ascension
3. Declination
4. Range rate

KK is a key which indicates if the measurements being made are: (1) on-board measurements or (2) moon beacons.

NN is a key which is used to obtain constants out of the common S array.

COMPHQ Partial Derivatives

To perform the covariance matrix updating for on-board observations and moon based beacons, a number of partial derivatives are required. The partials are derived in the following paragraphs.

The following quantities and relationships are used in the derivations. The vehicle's postion, $\vec{X}$, and velocity, $\vec{\dot{X}}$, relative to the celestial body or moon beacon are obtained from common. These vectors may be written as:

$$\vec{X} = \begin{pmatrix} X_1 \\ X_2 \\ X_3 \end{pmatrix} = -\vec{X}_V + \vec{X}_B = \begin{pmatrix} -X_V & +X_B \\ -Y_V & +Y_B \\ -Z_V & +Z_B \end{pmatrix}$$

$$\vec{\dot{X}} = \begin{pmatrix} \dot{X}_1 \\ \dot{X}_2 \\ \dot{X}_3 \end{pmatrix} = -\vec{\dot{X}}_V \; +\vec{\dot{X}}_B$$

Two relationships which will be used in the derivation are:

$$\frac{\partial PAR}{\partial \vec{X}} = - \frac{\partial PAR}{\partial \vec{X}_V}$$

$$\frac{\partial PAR}{\partial \dot{\vec{X}}} = - \frac{\partial PAR}{\partial \dot{\vec{X}}_V}$$

The FORTRAN program names for the above vector are as follows:

FORTRAN NAME	DIMENSION	DERIVATION NAME	DIMENSION
XREL	(3)	$\vec{X}$	(3)
VREL	(3)	$\dot{\vec{X}}$	(3)

The quantities being measured are the following:

$$R = \text{RANGE} = |\vec{X}| = (\vec{X}\cdot\vec{X})^{1/2}$$

$$R = \text{Range Rate} = \frac{d}{dt}|\vec{X}| = \frac{\vec{X}\cdot\vec{X}}{R}$$

$$RA = \tan^{-1}\left\{\frac{X_2}{X_1}\right\}$$

$$DEC = \sin^{-1}\left\{\frac{X_3}{R}\right\}$$

A. The following is a derivation of the partials relating the types of measurements and the vehicle state. In the program, these partials are used as a row vector. The FORTRAN name of the vector is H. H may be written as follows:

$$H = \left(\frac{\partial MEAS}{\partial X_V};\ \frac{\partial MEAS}{\partial Y_V};\ \frac{\partial MEAS}{\partial Z_V};\ \frac{\partial MEAS}{\partial \dot{X}_V};\ \frac{\partial MEAS}{\partial \dot{Y}_V};\ \frac{\partial MEAS}{\partial \dot{Z}_V}\right)$$

(1) Derivation of partials for range measurement.

$$R = (\vec{X}\cdot\vec{X})^{1/2}$$

$$H_R = \left(\frac{\partial R}{\partial \vec{X}_V} ; \frac{\partial R}{\partial \dot{\vec{X}}_V} \right) = \left(- \frac{X_1}{R}; - \frac{X_2}{R}; - \frac{X_3}{R}; 0; 0; 0 \right)$$

(2) Derivation of partials for range rate measurement.

$$\dot{R} = \frac{\vec{X}\cdot\dot{\vec{X}}}{R}$$

$$H_{RR} = \left(\frac{\partial \dot{R}}{\partial \vec{X}_V}; \frac{\partial \dot{R}}{\partial \dot{\vec{X}}_V} \right) = \left(- \frac{\dot{X}_1}{R} + \frac{\dot{R}}{R^2} X_1; - \frac{\dot{X}_2}{R} + \frac{\dot{R}}{R^2} X_2; - \frac{\dot{X}_3}{R} + \frac{\dot{R}}{R^2} X_3; \right.$$

$$\left. \frac{X_1}{R}; \frac{X_2}{R}; \frac{X_3}{R} \right)$$

(3) Derivation of partials for right ascension measurement.

$$RA = \mathrm{Tan}^{-1} \left(\frac{X_2}{X_1} \right) \equiv \mathrm{Tan}^{-1} \gamma$$

$$\frac{\partial RA}{\partial PAR} = (1 + \gamma^2)^{-1} \frac{\partial \gamma}{\partial PAR}$$

$$(1 + \gamma^2)^{-1} = \frac{X_1^{\,2}}{X_1^{\,2} + X_2^{\,2}}$$

$$H_{RA} = \left(\frac{\partial RA}{\partial \vec{X}_V}; \frac{\partial RA}{\partial \dot{\vec{X}}_V} \right) = \left(\frac{X_2}{X_1^{\,2} + X_2^{\,2}}; \frac{X_1}{X_1^{\,2} + X_2^{\,2}}; 0; 0; 0; 0 \right)$$

(4) Derivation of partials for declination measurement.

$$DEC = \sin^{-1}\left(\frac{X_3}{R}\right) \equiv \sin^{-1}\gamma$$

$$\frac{\partial DEC}{\partial PAR} = [1 - \gamma^2]^{-1/2} \quad \frac{\partial \gamma}{\partial PAR}$$

$$[1 - \gamma^2]^{-1/2} = \frac{R}{\sqrt{X_1^2 + X_2^2}}$$

$$H_{DEC} = \left(\frac{\partial DEC}{\partial \vec{X}_V}; \frac{\partial DEC}{\partial \dot{\vec{X}}_V}\right) = \left(\frac{X_1 X_3}{\sqrt{X_1^2 + X_2^2}\,R^2}; \quad \frac{X_2 X_3}{R^2\sqrt{X_1^2 + X_2^2}}; \quad \frac{\sqrt{X_1^2 + X_2^2}}{R^2}; \right.$$

$$\left. 0; 0; 0\right)$$

B. In order to include errors in the measurements being made due to station location errors and time bias, the partials of the measurements with respect to latitude, longitude, altitude, and time are required. In the program, the station location errors are used as a row vector with FORTRAN name DUM. DUM may be written as follows:

$$DUM = \left(\frac{\partial MEAS}{\partial LAT}; \frac{\partial MEAS}{\partial LON}; \frac{\partial MEAS}{\partial ALT}\right)$$

The following matrices are obtained from MONBTR and used in the computation of DUM.

$$STPARS = \begin{pmatrix} \frac{\partial X_1}{\partial LAT} & \frac{\partial X_1}{\partial LON} & \frac{\partial X_1}{\partial ALT} \\ \frac{\partial X_2}{\partial LAT} & \frac{\partial X_2}{\partial LON} & \frac{\partial X_2}{\partial ALT} \\ \frac{\partial X_3}{\partial LAT} & \frac{\partial X_3}{\partial LON} & \frac{\partial X_3}{\partial ALT} \end{pmatrix} \qquad STPARD = \begin{pmatrix} \frac{\partial \dot{X}_1}{\partial LAT} & \frac{\partial \dot{X}_1}{\partial LON} & \frac{\partial \dot{X}_1}{\partial ALT} \\ \frac{\partial \dot{X}_2}{\partial LAT} & \frac{\partial \dot{X}_2}{\partial LON} & \frac{\partial \dot{X}_2}{\partial ALT} \\ \frac{\partial \dot{X}_3}{\partial LAT} & \frac{\partial \dot{X}_3}{\partial LON} & \frac{\partial \dot{X}_3}{\partial ALT} \end{pmatrix}$$

(1) Derivation of range measurement error partials.

$$\frac{\partial R}{\partial LAT} = \sum_{i=1}^{3} \frac{\partial R}{\partial X_i} \frac{\partial X_i}{\partial LAT}$$

$$\frac{\partial R}{\partial LAT} = \sum_{i=1}^{3} -H_R(1,i)\ \text{STPARS}(i,1)$$

Similarly

$$\frac{\partial R}{\partial LON} = \sum_{i=1}^{3} -H_R(1,i)\ \text{STPARS}(i,2)$$

$$\frac{\partial R}{\partial ALT} = \sum_{i=1}^{3} -H_R(1,i)\ \text{STPARS}(i,3)$$

(2) Derivation of range rate measurement error partials.

$$\frac{\partial \dot{R}}{\partial LAT} = \sum_{i=1}^{3} \frac{\partial \dot{R}}{\partial X_i} \frac{\partial X_i}{\partial LAT} + \frac{\partial R}{\partial \dot{X}_i} \frac{\partial \dot{X}_i}{\partial LAT}$$

$$\frac{\partial \dot{R}}{\partial LAT} = \sum_{i=1}^{3} -H_{RR}(1,i)\ \text{STPARS}(i,1) - H_{RR}(1,i+3)\ \text{STPARD}(i,1)$$

Similarly

$$\frac{\partial \dot{R}}{\partial LON} = \sum_{i=1}^{3} -H_{RR}(1,i)\ \text{STPARS}(i,2) - H_{RR}(1,i+3)\ \text{STPARD}(i,2)$$

$$\frac{\partial \dot{R}}{\partial ALT} = \sum_{i=1}^{3} -H_{RR}(1,i)\ STPARS(i,3) - H_{RR}(1,i+3)\ STPARD(i,3)$$

(3) Derivation of right ascension measurement error partials.

$$\frac{\partial RA}{\partial LAT} = \sum_{i=1}^{3} \frac{\partial RA}{\partial X_i} \frac{\partial X_i}{\partial LAT}$$

$$\frac{\partial RA}{\partial LAT} = \sum_{i=1}^{3} -H_{RA}(1,i)\ STPARS(i,1)$$

Similarly

$$\frac{\partial RA}{\partial LON} = \sum_{i=1}^{3} -H_{RA}(1,i)\ STPARS(i,2)$$

$$\frac{\partial RA}{\partial ALT} = \sum_{i=1}^{3} -H_{RA}(1,i)\ STPARS(i,3)$$

(4) Derivation of declination measurement error partials.

$$\frac{\partial DEC}{\partial LAT} = \sum_{i=1}^{3} \frac{\partial DEC}{\partial X_i} \frac{\partial X_i}{\partial LAT}$$

$$\frac{\partial DEC}{\partial LAT} = \sum_{i=1}^{3} -H_{DEC}(1,i)\ STPARS(i,1)$$

Similarly

$$\frac{\partial DEC}{\partial LON} = \sum_{i=1}^{3} -H_{DEC}(1,i)\ STPARS(i,2)$$

$$\frac{\partial DEC}{\partial ALT} = \sum_{i=1}^{3} -H_{DEC}(1,i)\ STPARS(i,3)$$

(5) Derivation of time derivatives of measurement quantities to permit inclusion of time bias errors.

$$\frac{\partial R}{\partial t} = \dot{R} = \frac{\vec{X}\cdot\dot{\vec{X}}}{R}$$

$$\frac{\partial RA}{\partial t} = \sum_{i=1}^{3} \frac{\partial RA}{\partial X_i}\ \frac{\partial X_i}{\partial t}$$

$$\frac{\partial RA}{\partial t} = \sum_{i=1}^{3} -H_{RA}(1,i)\ \dot{X}_i$$

Similarly

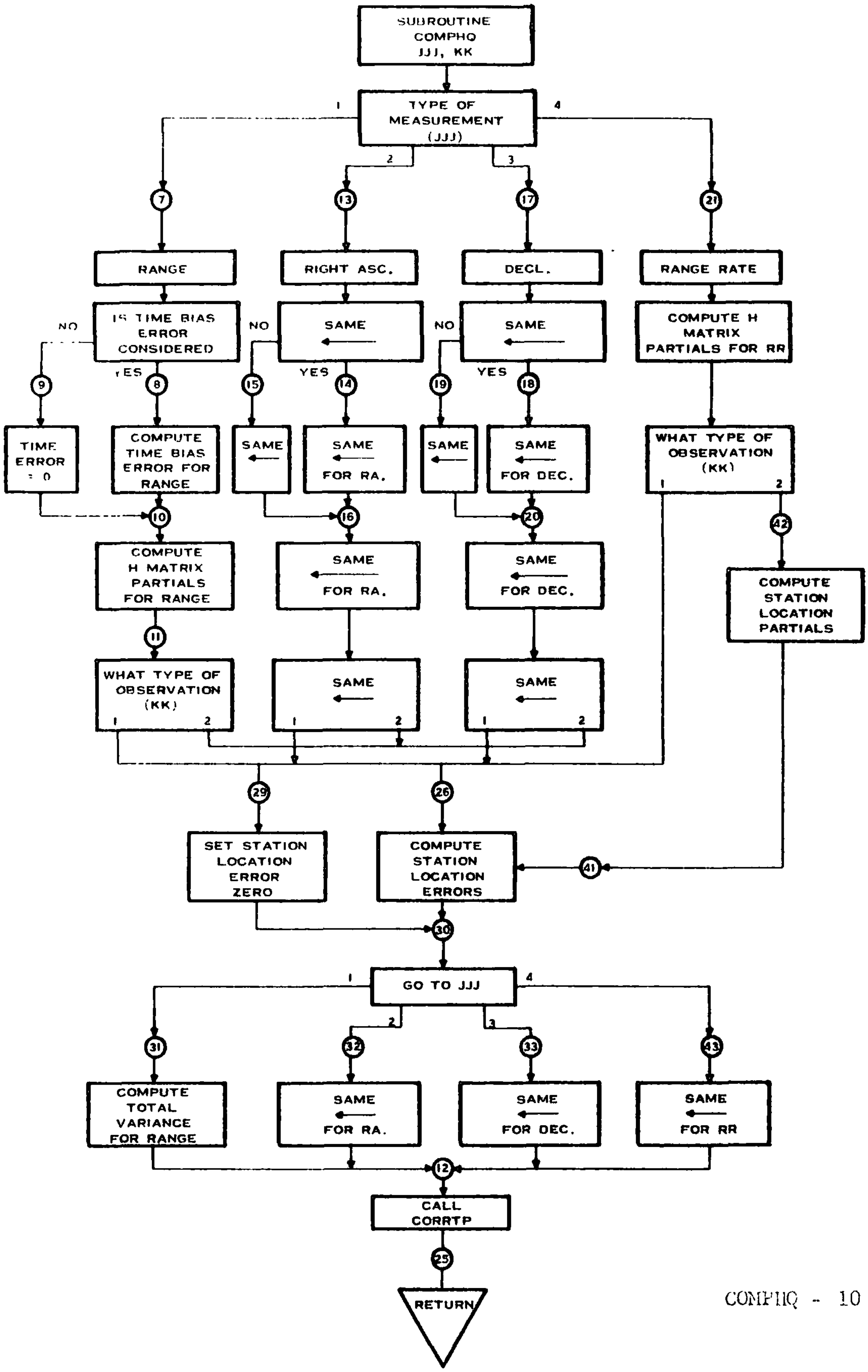
$$\frac{\partial DEC}{\partial t} = \sum_{i=1}^{3} -H_{DEC}(1,i)\ \dot{X}_i$$

The errors in the measurements are computed in the following manner for station location errors and time bias.

$$\underset{1x1}{QQSTAT} = \underset{1x3}{\left(DUM_{MEAS}\right)^2} \underset{3x1}{\begin{pmatrix} \sigma^2 LAT \\ \sigma^2 LON \\ \sigma^2 ALT \end{pmatrix}} \begin{matrix} \text{Station} \\ \text{Location} \\ \text{Error} \end{matrix}$$

$$QQDOT = \left(\frac{\partial MEAS}{\partial t} \cdot \sigma_{TIME} \right)^2 \qquad \text{Time Bias Error}$$

The computation of the H matrices and the measurement errors are the primary computations in COMPHQ. COMPHQ calls CORRTP which uses quantities to update the state covariance matrix.


SUBROUTINE COMPHQ JJJ, KK
TYPE OF MEASUREMENT (JJJ)
1
2
3
4
7
13
17
21
RANGE
RIGHT ASC.
DECL.
RANGE RATE
IS TIME BIAS ERROR CONSIDERED
NO
YES
SAME
SAME
COMPUTE H MATRIX PARTIALS FOR RR
9
8
15
14
19
18
TIME ERROR = 0
COMPUTE TIME BIAS ERROR FOR RANGE
SAME
SAME FOR RA.
SAME
SAME FOR DEC.
WHAT TYPE OF OBSERVATION (KK)
10
16
20
42
COMPUTE H MATRIX PARTIALS FOR RANGE
SAME FOR RA.
SAME FOR DEC.
COMPUTE STATION LOCATION PARTIALS
11
WHAT TYPE OF OBSERVATION (KK)
SAME
SAME
29
26
SET STATION LOCATION ERROR ZERO
COMPUTE STATION LOCATION ERRORS
41
30
GO TO JJJ
31
32
33
43
COMPUTE TOTAL VARIANCE FOR RANGE
SAME FOR RA.
SAME FOR DEC.
SAME FOR RR
12
CALL CORRTP
25
RETURN

COMPHQ - 10

```
*       LABEL
*       SYMBOL TABLE
CEC2008  SUBROUTINE COMPHQ
        SUBROUTINE COMPHQ(JJJ,KK,NN)
        COMMON T,S,C,IC
        DIMENSION T(1360),S(1000),C(1000),IC(1)
       1,XREL(3),VREL(3),H(6),P(6,6),STPARS(3,3),DUM(3),OUTPUT(6)
        DIMENSION STPARD(3,3)
        EQUIVALENCE (T,TDUM),(S,SDUM),(C,CDUM),(IC,ICDUM)
       1,(C(895),XREL),(C(898),VREL),(C(894), OBNO),(C(893),XMAG),
       2(C(892),DEX),(C(891),DEN1),(C(890),DEN2),(C(788),STPARS)
       3,(C(889),DEX2),(C(888),DEX3),(C(652),P),(C(973),OUTPUT)
       4,(S(73),RTD)
        EQUIVALENCE (C(800),STPARD)
C       KK  1=ONBOARD MEASUREMENTS   2=MOON BEACONS
C       JJJ  1=RANGE  2=RIGHT ASC.  3=DECL.  4=RANGE RATE
        NN=NN
        JJJ=JJJ
        KK=KK
        NNN=NN/5+657
        GO TO (7,13,17,21),JJJ
      7 CONTINUE
        IF(S(NN+8))9,9,8
      8 CONTINUE
C       GET HERE TO COMPUTE ERROR IN RANGE OBS. DUE TO TIME BIAS
        RDOT=DOT(XREL,VREL)*DEX3
C       RDOT=RANGE RATE
        OUTPUT(2)=RDOT
        RDOT2=RDOT*RDOT
        QQRDOT=RDOT2*S(NN+8)
C       QQRDOT=RANGE VARIANCE DUE TO TIME BIAS
        GO TO 10
      9 CONTINUE
        QQRDOT=0.
     10 CONTINUE
C       THE FOLLOWING ARE H MATRIX CALCULATIONS FOR RANGE
        DO 11 I=1,3
        K=I+3
        H(I)=-XREL(I)*DEX3
        H(K)=0.
     11 CONTINUE
        GO TO (29,26),KK
   31   CONTINUE
        QQ=S(NN)/OBNO+S(NN+4)+QQRDOT+S(425)*XMAG**2+QQSTAT
C       QQ=TOTAL RANGE VARIANCE=INSTR.+BIAS+TIME BIAS ERROR+SPEED LIGHT
C       +STATION LOCATION ERRORS
     12 CONTINUE
        CALL CORRTP (QQ,H,P)
C       CORRTP UPDATES COVARIANCE MATRIX FOR MEASUREMENT
        GO TO 25
     13 CONTINUE
```

```
      IF(S(NN+8))15,15,14                                               COMP0440
   14 CONTINUE                                                          COMP0450
C     GET HERE TO COMPUTE ERROR IN RA MEASUREMENT DUE TO CLOCK BIAS     COMP0460
      RADOT=(VREL(2)*XREL(1)-VREL(1)*XREL(2))*DEN2                      COMP0470
      OUTPUT(4)=RADOT*RTD                                               COMP0475
C     RADOT=RIGHT ASCENSION RATE                                        COMP0480
      RADOT2=RADOT*RADOT                                                COMP0490
      QQRADT=RADOT2*S(NN+8)                                             COMP0500
C     QQRADT=RIGHT ASCENSION VARIANCE DUE TO TIME BIAS                  COMP0510
      GO TO 16                                                          COMP0520
   15 CONTINUE                                                          COMP0530
      QQRADT=0.                                                         COMP0540
   16 CONTINUE                                                          COMP0550
C     THE FOLLOWING ARE H MATRIX CALCULATIONS FOR RA                    COMP0560
      H(6)=0.                                                           COMP0570
      H(5)=0.                                                           COMP0580
      H(4)=0.                                                           COMP0590
      H(3)=0.                                                           COMP0600
      H(1)= XREL(2)*DEN2                                                COMP0610
      H(2)=-XREL(1)*DEN2                                                COMP0620
      GO TO (29,26),KK                                                  COMP0630
  32  CONTINUE                                                          COMP0640
      QQ=S(NN+1)/OBNO+S(NN+5)+QQRADT+QQSTAT                             COMP0650
C     QQ=TOTAL RIGHT ASCENSION VARIANCE                                 COMP0660
      GO TO 12                                                          COMP0670
   17 CONTINUE                                                          COMP0680
      IF(S(NN+8))19,19,18                                               COMP0690
   18 CONTINUE                                                          COMP0700
C     GET HERE TO COMPUTE ERROR IN DEC.MEAS.DUE TO CLOCK BIAS           COMP0710
      CRUD=XREL(1)*VREL(1)+XREL(2)*VREL(2)                              COMP0720
      CRUD2=VREL(3)*DEX-XREL(3)*CRUD                                    COMP0730
      DEDOT=CRUD2/(SQRTF(DEX)*DEX2)                                     COMP0740
      OUTPUT(6)=DEDOT*RTD                                               COMP0745
      DEDOT2=DEDOT*DEDOT                                                COMP0750
      QQDEDT=DEDOT2*S(NN+8)                                             COMP0760
C     QQDEDT=DECLINATION VARIANCE DUE TO TIME BIAS                      COMP0770
      GO TO 20                                                          COMP0780
   19 CONTINUE                                                          COMP0790
      QQDEDT=0.                                                         COMP0800
   20 CONTINUE                                                          COMP0810
C     THE FOLLOWING ARE H MATRIX COMP. FOR DECLINATION                  COMP0820
      H(4)=0.                                                           COMP0830
      H(5)=0.                                                           COMP0840
      H(6)=0.                                                           COMP0850
      H(1)= XREL(1)*XREL(3)*DEN1                                        COMP0860
      H(2)= XREL(2)*XREL(3)*DEN1                                        COMP0870
      H(3)=-DEX*DEN1                                                    COMP0880
      GO TO (29,26),KK                                                  COMP0890
  33  CONTINUE                                                          COMP0900
      QQ=S(NN+2)/OBNO+S(NN+6)+QQDEDT+QQSTAT                             COMP0910
C     QQ=TOTAL DECLINATION VARIANCE                                     COMP0920
```

```
      GO TO 12                                                          COMP093
   21 CONTINUE                                                          COMP094
C     THE FOLLOWING ARE H MATRIX COMP. FOR RANGE RATE                   COMP095
      RDOT=DOT(XREL,VREL)*DEX3                                          COMP096
      OUTPUT(2)=RDOT                                                    COMP096
      DO 22 J=1,3                                                       COMP097
      K=J+3                                                             COMP098
      H(K)=-XREL(J)*DEX3                                                COMP099
      H(J)=-DEX3*(VREL(J)+H(K)*RDOT)                                    COMP100
   22 CONTINUE                                                          COMP100
      GO TO (29,42),KK                                                  COMP100
   42 CONTINUE                                                          COMP100
      DO 40 I=1,3                                                       COMP100
      DUM(I)=0.                                                         COMP101
      DO 40 J=1,3                                                       COMP101
      JJ=J+3                                                            COMP101
   40 DUM(I)=DUM(I)+H(J)*STPARS(J,I)+H(JJ)*STPARD(J,I)                  COMP101
      GO TO 41                                                          COMP102
   43 CONTINUE                                                          COMP102
      QQ=S(NN+3)/OBNO+S(NN+7)+QQSTAT                                    COMP102
C     QQ=RANGE RATE VARIANCE                                            COMP103
      GO TO 12                                                          COMP104
   26 CONTINUE                                                          COMP105
C     THE FOLLOWING COMPUTATIONS ARE FOR STATION LOCATION ERRORS        COMP106
      DO 27 I=1,3                                                       COMP107
      DUM(I)=0.                                                         COMP108
      DO 27 J=1,3                                                       COMP109
      DUM(I)=DUM(I)+H(J)*STPARS(J,I)                                    COMP110
   27 CONTINUE                                                          COMP111
   41 CONTINUE                                                          COMP111
      QQSTAT=0.                                                         COMP112
      DO 28 I=1,3                                                       COMP113
      L=NNN+I                                                           COMP114
      QQSTAT=QQSTAT+(DUM(I)**2)*S(L)                                    COMP115
   28 CONTINUE                                                          COMP116
      GO TO 30                                                          COMP117
   29 CONTINUE                                                          COMP118
      QQSTAT=0.                                                         COMP119
   30 CONTINUE                                                          COMP120
      GO TO(31,32,33,43),JJJ                                            COMP121
   25 CONTINUE                                                          COMP122
      RETURN                                                            COMP123
      END                                                               COMP
```

Page Intentionally Left Blank

Subroutine: CONST1

Purpose: To read permanent or semipermanent constants into the common S array.

Calling Sequence:

CALL CONST1

Input and **Output**

Common storages used or required:	T, S
Subroutines required:	None
Functions required:	None
Approximate number of storages required:	669 DEC

```
*      LABEL
*      SYMBOL TABLE
CEC2026  SUBROUTINE CONST1
       SUBROUTINE  CONST1
       COMMON T,S
       DIMENSION T(1360),S(1000)
       EQUIVALENCE (T,TDUM),(S,SDUM)
C      PLANETARY MASSES 1-6,
C      PLANETARY MASSES 1-6 ,ORDER EARTH,MOON,SUN,VENUS,MARS,JUPITER
C      -KM**3/SEC2
       S(1)=398603.2
       S(2)=4900.7588
       S(3)=.13271545E12
       S(4)=324769.50
       S(5)=42915.515
       S(6)=.12671059E9
C      6-9 ARE SPARES
C      10-19 ARE EARTH CONSTANTS,RATE,J,H,D,EQUAT RAD,POLAR RAD
C      RAD/SEC
       S(10)=7.2921152E-5
       S(11)=.162345E-2
       S(12)=-.575E-5
       S(13)=.7875E-5
       S(14)=6378.2064
       S(15)=6356.5838
C      20-29 ARE MOON CONSTANTS ,G,A,B,C,RAD
       S(20)=.6671E-19
       S(21)=.88746E29
       S(22)=.88764E29
       S(23)=.88801E29
       S(24)=1738.
C      30-59 SUN CONSTANTS
C      40-49 VENUS CONSTANTS
C      50-59 MARS CONSTANTS
C      60-69 JUPITER CONSTANTS
C CONVERSION FACTOR DAYS TO SECONDS
       S(70)=86400.
C CONVERSION FACTOR SECONDS TO DAYS
       S(71)=1./S(70)
C CONVERSION FACTOR FOR DEGREES TO RADIANS
       S(72)=.017453296
C CONVERSION FACTOR FOR RADIANS TO DEGREES
       S(73)=57.29578
       S(80)=28.48713
       S(81)=279.42315
       S(82)=.00257
       S(84)=90.
C      111 IS TYPE OF INPUT(1)EQUATOR 1950 (2)EQUATOR OF DATE(3)
C      EARTH-FIXED
C      (4),(5),(6) ARE SPHERICAL INPUT IN, RESPECTIVELY, (1),(2),(3)
C      112 AND 113 ARE(YEAR MONTH.DAY)(HOUR MIN.SEC) OF INJECTION DATE
```

```
C      114 IS CENTRAL BODY NUMBER,(1)EARTH,(2)MOON(3)SUN(4)VENUS(5)        CONS04
C      MARS(6)JUPITER                                                       CONS04
C      115-120 ARE INPUT STATE CORRESPONDING TO S(111) THRU 114.           CONS05
C      121  TARGET BODY                                                     CONS05
C 122 TYPE OF OUTPUT;KOUT                                                   CONS05
       S(121)=2.                                                            CONS05
C 123 TSECO                                                                 CONS05
C N GOES FROM 0 TO 19                                                       CONS05
C(125)+N*15  VARIANCE OF RANGE                    KM**2                     CONS05
C(126)+N*15  VARIANCE OF AZIMUTH                  RAD**2                    CONS05
C(127)+N*15  VARIANCE OF ELEVATION                RAD**2                    CONS05
C(128)+N*15  VARIANCE OF RANGE RATE               KM**2/SEC**2              CONS05
C(129)+N*15  VARIANCE OF LATITUDE                 RAD**2                    CONS06
C(130)+N*15  VARIANCE OF LONGITUDE                RAD**2                    CONS06
C(131)+N*15  VARIANCE OF ALTITUDE                 KM**2                     CONS06
C(132)+N*15  VARIANCE OF AZIMUTH BIASES           RAD**2                    CONS06
C(133)+N*15  VARIANCE OF ELEVATION BIASES         RAD**2                    CONS06
C(134)+N*15  LATITUDE OF STATION                  DEG                       CONS06
C(135)+N*15  LONGTUDE OF STATION                  DEG                       CONS06
C(136)+N*15  ALTITUDE OF STATION                  KM                        CONS06
C(137)+N*15  STATION NAME                                                   CONS06
C(138)+N*15  PERIOD OF OBSERVATION                SEC                       CONS06
C(139)+N*15  VARIANCE OF TIME BIAS                SEC**2                    CONS07
C      STATION 1 ANTIGUA RADAR 91.1                                         CONS07
       S(125)=.000225                                                       CONS07
       S(126)=.000004                                                       CONS07
       S(127)=.000004                                                       CONS07
       S(128)=.00000001                                                     CONS07
       S(129)=.0000000001                                                   CONS07
       S(130)=.0000000001                                                   CONS07
       S(131)=.0001                                                         CONS07
       S(134)=17.0267                                                       CONS07
       S(135)=298.2247                                                      CONS08
       S(136)=0.                                                            CONS08
       S(137)=6HANTIG                                                       CONS08
       S(138)=1.                                                            CONS08
C      STATION 2 ASCENSION RADAR 12.16                                      CONS08
       S(140)=.000225                                                       CONS08
       S(141)=.000004                                                       CONS08
       S(142)=.000004                                                       CONS08
       S(143)=.00000001                                                     CONS08
       S(144)=.0000000001                                                   CONS08
       S(145)=.0000000001                                                   CONS09
       S(146)=.0001                                                         CONS09
       S(149)=-7.898                                                        CONS09
       S(150)=345.587393                                                    CONS09
       S(151)=0.                                                            CONS09
       S(152)=6HASCENS                                                      CONS09
       S(153)=1.                                                            CONS09
C      STATION 3 MILLSTONE HILL RADAR                                       CONS09
       S(155)=.000225                                                       CONS09
```

```
      S(156)=.000004                                          CONS0990
      S(157)=.000004                                          CONS1000
      S(158)=.00000001                                        CONS1010
      S(159)=.0000000001                                      CONS1020
      S(160)=.0000000001                                      CONS1030
      S(161)=.0001                                            CONS1040
      S(164)=42.4232                                          CONS1050
      S(165)=288.5080                                         CONS1060
      S(166)=0.                                               CONS1070
      S(167)=6HMILHIL                                         CONS1080
      S(168)=1.                                               CONS1090
C     STATION 4 MOBILE STATION MTS                            CONS1100
      S(170)= 2500.E-6                                        CONS1110
      S(171)= .04 E-6                                         CONS1120
      S(172)= .04 E-6                                         CONS1130
      S(173)= .04 E-6                                         CONS1140
      S(174)= 1. E-10                                         CONS1150
      S(175)= 1. E-10                                         CONS1160
      S(176)= 9. E-6                                          CONS1170
      S(177)= .0016E-6                                        CONS1180
      S(178)= .0016E-6                                        CONS1190
      S(179)=-25.88472                                        CONS1200
      S(180)= 27.70528                                        CONS1210
      S(181)= 1.54302                                         CONS1220
      S(182)=6HMOBMTS                                         CONS1230
      S(183)= 30.                                             CONS1240
      S(184)= 9.E-6                                           CONS1250
C     STATION 5 AMR TRACKER GE                                CONS1260
      S(185)=.000225                                          CONS1270
      S(186)=.000004                                          CONS1280
      S(187)=.000004                                          CONS1290
      S(188)=.00000001                                        CONS1300
      S(189)=.0000000001                                      CONS1310
      S(190)=.0000000001                                      CONS1320
      S(191)=.0001                                            CONS1330
      S(194)=28.278103                                        CONS1340
      S(195)=279.418297                                       CONS1350
      S(196)=0.                                               CONS1360
      S(197)=6HGEAMR                                          CONS1370
      S(198)=1.                                               CONS1380
C     STATION 6 BERMUDA                                       CONS1390
      S(200)=.000225                                          CONS1400
      S(201)=.000004                                          CONS1410
      S(202)=.000004                                          CONS1420
      S(203)=.00000001                                        CONS1430
      S(204)=.0000000001                                      CONS1440
      S(205)=.0000000001                                      CONS1450
      S(206)=.0001                                            CONS1460
      S(209)=32.160973                                        CONS1470
      S(210)=295.299179                                       CONS1480
      S(211)=0.                                               CONS1490
```

```
      S(212)=6HBERMUD                                                   CONS15
      S(213)=1.                                                         CONS15
C     STATION 7 GOLDSTONE RECEIVER                                      CONS15
C     50M SIG                                                           CONS15
      S(215) =2500.E-6                                                  CONS15
C     .2MIL                                                             CONS15
      S(216) =.04 E-6                                                   CONS15
C     .2MIL                                                             CONS15
      S(217) =.04 E-6                                                   CONS15
C     .2M/SEC                                                           CONS15
      S(218) =.04 E-6                                                   CONS16
C      63.78M                                                           CONS16
      S(219) =1.E-10                                                    CONS16
C      63.78M                                                           CONS16
      S(220) =1.E-10                                                    CONS16
C     3M                                                                CONS16
      S(221) =9.E-6                                                     CONS16
C     .04MR                                                             CONS16
      S(222) =.0016E-6                                                  CONS16
C     .04MR                                                             CONS16
      S(223) =.0016E-6                                                  CONS17
      S(224) =35.3895                                                   CONS17
      S(225) =243.15175                                                 CONS17
      S(226) =1.03754                                                   CONS17
      S(227)=6HGOSTRC                                                   CONS17
      S(228) =30.                                                       CONS17
      S(229) =9.E-6                                                     CONS17
C     STATION 8 GOLDSTONE TRANSMITTER                                   CONS17
      S(230)=.000225                                                    CONS17
      S(231)=.000004                                                    CONS17
      S(232)=.000004                                                    CONS18
      S(233)=.00000001                                                  CONS18
      S(234)=.0000000001                                                CONS18
      S(235)=.0000000001                                                CONS18
      S(236)=.0001                                                      CONS18
      S(239) =35.29985                                                  CONS18
      S(240) =243.19539                                                 CONS18
      S(241) =.98949                                                    CONS18
      S(242)=6HGOSTTR                                                   CONS18
      S(243)=1.0                                                        CONS18
C     STATION 9 GBI RADAR XN-2                                          CONS19
      S(245)=.000225                                                    CONS19
      S(246)=.000004                                                    CONS19
      S(247)=.000004                                                    CONS19
      S(248)=.00000001                                                  CONS19
      S(249)=.0000000001                                                CONS19
      S(250)=.0000000001                                                CONS19
      S(251)=.0001                                                      CONS19
      S(254)=26.459736                                                  CONS19
      S(255)=281.651892                                                 CONS19
      S(256)=0.                                                         CONS20
```

```
      S(257)=6HGBIRAD                                                   CONS2010
      S(258)=1.0                                                        CONS2020
C     STATION 10 JOHANNESBURG                                           CONS2030
C     50M SIG                                                           CONS2040
      S(260) =2500.E-6                                                  CONS2050
C     .2MIL                                                             CONS2060
      S(261) =.04 E-6                                                   CONS2070
C     .2MIL                                                             CONS2080
      S(262) =.04 E-6                                                   CONS2090
C     .2M/SEC                                                           CONS2100
      S(263) =.04 E-6                                                   CONS2110
C      63.78M                                                           CONS2120
      S(264) =1.E-10                                                    CONS2130
C      63.78M                                                           CONS2140
      S(265) =1.E-10                                                    CONS2150
C     3M SIG                                                            CONS2160
      S(266) = 9.E-6                                                    CONS2170
C     .04MR                                                             CONS2180
      S(267) = .0016E-6                                                 CONS2190
C     .04MR                                                             CONS2200
      S(268) = .0016E-6                                                 CONS2210
C      DEG                                                              CONS2220
      S(269) =-25.88735                                                 CONS2230
C      DEG                                                              CONS2240
      S(270) = 27.68478                                                 CONS2250
C      KM                                                               CONS2260
      S(271) = 1.38192                                                  CONS2270
      S(272)=6HJOHABG                                                   CONS2280
C      SEC                                                              CONS2290
      S(273) = 30.                                                      CONS2300
C     3MS SIG                                                           CONS2310
      S(274) = 9.E-6                                                    CONS2320
C     STATION IN HAWAII                                                 CONS2330
      S(275)=.000225                                                    CONS2340
      S(276)=.000004                                                    CONS2350
      S(277)=.000004                                                    CONS2360
      S(278)=.00000001                                                  CONS2370
      S(279)=.000000001                                                 CONS2380
      S(280)=.000000001                                                 CONS2390
      S(281)=.0001                                                      CONS2400
      S(284)=18.823066                                                  CONS2410
      S(285)=204.314722                                                 CONS2420
      S(286)=0.                                                         CONS2430
      S(287)=6HHAWAII                                                   CONS2440
      S(288)=1.0                                                        CONS2450
C     STATION 12 JODRELL BANK                                           CONS2460
      S(290)=.000225                                                    CONS2470
      S(291)=.000004                                                    CONS2480
      S(292)=.000004                                                    CONS2490
      S(293)=.00000001                                                  CONS2500
      S(294)=.000000001                                                 CONS2510
```

```
      S(295)=.0000000001                                    CONS25
      S(296)=.0001                                          CONS25
      S(299)=53.049636                                      CONS25
      S(300)=357.693889                                     CONS25
      S(301)=0.                                             CONS25
      S(302)=6HJODBNK                                       CONS25
      S(303)=1.0                                            CONS25
C     STATION 13 PUERTO RICO RADAR 9.1                      CONS25
      S(305)=.000225                                        CONS26
      S(306)=.000004                                        CONS26
      S(307)=.000004                                        CONS26
      S(308)=.00000001                                      CONS26
      S(309)=.0000000001                                    CONS26
      S(310)=.0000000001                                    CONS26
      S(311)=.0001                                          CONS26
      S(314)=18.060396                                      CONS26
      S(315)=292.911805                                     CONS26
      S(316)=0.                                             CONS26
      S(317)=6HPURICO                                       CONS27
      S(318)=1.0                                            CONS27
C     STATION 14 SAN SALVADORE                              CONS27
      S(320)=.000225                                        CONS27
      S(321)=.000004                                        CONS27
      S(322)=.000004                                        CONS27
      S(323)=.00000001                                      CONS27
      S(324)=.0000000001                                    CONS27
      S(325)=.0000000001                                    CONS27
      S(326)=.0001                                          CONS27
      S(329)=23.173566                                      CONS28
      S(330)=285.4956                                       CONS28
      S(331)=0.                                             CONS28
      S(332)=6HSANSAL                                       CONS28
      S(333)=1.0                                            CONS28
C     STATION 15 WOOMERA                                    CONS28
C     50W SIG                                               CONS28
      S(335) = 2500.E-6                                     CONS28
C     .2MIL                                                 CONS28
      S(336) = .04E-6                                       CONS28
C     .2MIL                                                 CONS29
      S(337) = .04E-6                                       CONS29
C     .2M/SEC                                               CONS29
      S(338) = .04E-6                                       CONS29
C     63.78M                                                CONS29
      S(339) = 1.E-10                                       CONS29
C     63.78M                                                CONS29
      S(340) = 1.E-10                                       CONS29
C      3M                                                   CONS29
      S(341) = 9.E-6                                        CONS29
C     .04MR                                                 CONS30
      S(342) = .0016E-6                                     CONS30
C     .04MR                                                 CONS30
```

```
      S(343) = .0016E-6                                  CONS3030
      S(344) =-31.38287                                  CONS3040
      S(345) =136.88502                                  CONS3050
      S(346) = .15079                                    CONS3060
      S(347)=6HWOMERA                                    CONS3070
C     SEC                                                CONS3080
      S(348) = 30.                                       CONS3090
C     3MS                                                CONS3100
      S(349) = 9.E-6                                     CONS3110
C   STATION 16   JPL CAPE CANAVERAL                      CONS3120
      S(350)=.000225                                     CONS3130
      S(351)=.000004                                     CONS3140
      S(352)=.000004                                     CONS3150
      S(353)=.00000001                                   CONS3160
      S(354)=.0000000001                                 CONS3170
      S(355)=.0000000001                                 CONS3180
      S(356)=.0001                                       CONS3190
C     DEG                                                CONS3200
      S(359) = 28.48713                                  CONS3210
C     DEG                                                CONS3220
      S(360) = 279.42315                                 CONS3230
C     KM                                                 CONS3240
      S(361) = .00257                                    CONS3250
      S(362) = 6HCAPJET                                  CONS3260
      S(363)=1.0                                         CONS3265
C    425 =SPEED OF LIGHT ACCURACY                        CONS3270
C     STATION 17 MAJUNGA                                 CONS3271
      S(369)=.0000000001                                 CONS3272
      S(370)=.0000000001                                 CONS3273
      S(371)=.0001                                       CONS3274
      S(374)=-15.216666                                  CONS3275
      S(375)=46.38333                                    CONS3276
      S(377)=6HMAJUNG                                    CONS3277
      S(378)=30.                                         CONS3278
C     STATION 18 CARNARVON                               CONS3279
      S(384)=.0000000001                                 CONS3280
      S(385)=.0000000001                                 CONS3281
      S(386)=.0001                                       CONS3282
      S(389)=-24.8666                                    CONS3283
      S(390)=113.63333                                   CONS3284
      S(392)=6HCARNVN                                    CONS3285
      S(393)=30.                                         CONS3286
C     STATION 19 ROSMAN                                  CONS3287
      S(399)=.0000000001                                 CONS3288
      S(400)=.0000000001                                 CONS3289
      S(401)=.0001                                       CONS3290
      S(404)=35.20000                                    CONS3291
      S(405)=277.1333                                    CONS3292
      S(407)=6HROSMAN                                    CONS3293
      S(408)=30.                                         CONS3294
      S(425)=1.E-12                                      CONS3295
```

```
C   FROM S(426)  TO  S(470) IS ONBOARD ERROR DATA                                CONS330
      S(475)=1.0                                                                 CONS331
      S(476)=100.                                                                CONS332
      S(477)=200.                                                                CONS333
      S(478)=300.                                                                CONS334
      S(479)=400.                                                                CONS335
      S(480)=500.                                                                CONS336
      S(481)=600.                                                                CONS337
      S(485)=.03                                                                 CONS337
C  N GOES FROM 0 TO 9  LUNAR BASED BEACONS                                       CONS338
C (500)+N*15  VARIANCE OF RANGE                      KM2                         CONS339
C (501)+N*15  VARIANCE OF AZIMUTH                    RAD2                        CONS340
C (502)+N*15  VARIANCE OF ELEVATION                  RAD2                        CONS341
C (503)+N*15  VARIANCE OF RANGE RATE                 KM2/SEC2                    CONS342
C (504)+N*15  VARIANCE OF LATITUDE                   RAD2                        CONS343
C (505)+N*15  VARIANCE OF LONGITUDE                  RAD2                        CONS344
C (506)+N*15  VARIANCE OF ALTITUDE                   KM2                         CONS345
C (507)+N*15  VARIANCE OF AZIMUTH BIAS               RAD2                        CONS346
C (508)+N*15  VARIANCE OF ELEVATION BIAS             RAD2                        CONS347
C (509)+N*15  LATITUDE OF STATION                    DEG                         CONS348
C (510)+N*15  LONGITUDE OF STATION                   DEG                         CONS349
C (511)+N*15  ALTITUDE OF STATION                    KM                          CONS350
C (512)+N*15  STATION NAME                                                       CONS351
C (513)+N*15  PERIOD OF OBSERVATION                  SEC                         CONS352
C (514)+N*15  SPARE                                                              CONS353
C (650) TO (685)  IS INITIAL P MATRIX                                            CONS354
C (686) TO (721)     INITIAL RAR MATRIX                                          CONS355
C (722) TO (739)     TRANSITION MATRIX  FTA GUIDANCE                             CONS356
      S(900)=6378.165                                                            CONS357
      S(901)=1738.                                                               CONS358
      RETURN                                                                     CONS362
      END                                                                        CONS
```

Page Intentionally Left Blank

Subroutine: CONVPI

Purpose: Subroutine CONVPI transforms the input covariance matrices to the mean equinox of 1950 coordinate system. The input covariance matrices may be input in four coordinate systems:

1. Instantaneous launch pad tangent plane
2. Instantaneous injection tangent plane
3. True equinox of date
4. Mean equinox of 1950.

Calling Sequence:

CALL CONVPI (ITYPE, PIN, POUT)

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	ITYPE	1			Logic key which indicates the
					type of input coordinate frame
I	PIN	21		Km^2, $(Km/Sec)^2$	Upper half of input covariance
					matrix in vector form
0	POUT	6, 6	$E(\vec{X}_{50}\ \vec{X}_{50}^{\ T})$	Km^2, $(Km/sec)^2$	Output covariance matrix (1950)

Common storages used or required: T, S, C, IC

Subroutines required: CROSS, GHA, MULT, NUTAIT, PTRAN, ROTEQ, TIMED, TRAC

Functions required: COS SIN FNORM

Approximate number of storages required: 476 DEC

Calling Sequence Elaboration

The calling sequence consists of CALL CONVPI (INTYPE, PIN, POUT) where:

INTYPE is a logic key which indicates the type of coordinate system in which the input covariance matrix is expressed. The range of the key is one to five. One through four indicate the four coordinate systems described above. A five in the call sequence indicates that the input matrix is in the same coordinate frame as the input matrix of the preceding call of the subroutine.

PIN is in the input covariance matrix. The matrix must be in the form of a column matrix with dimension (21). The (21) elements represent the upper half of the input covariance matrix.

$$PIN = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ & 7 & 8 & 9 & 10 & 11 \\ & & 12 & 13 & 14 & 15 \\ & & & 16 & 17 & 18 \\ & & & & 19 & 20 \\ & & & & & 21 \end{bmatrix}$$

Input Elements

The required order of the input covariance matrix is the following:
For INTYPE = 1 or 2

$$PIN = E \begin{bmatrix} AV \\ NOR \\ ALT \\ \dot{AV} \\ \dot{NOR} \\ \dot{ALT} \end{bmatrix}_{6x1} \begin{bmatrix} AV \\ NOR \\ ALT \\ \dot{AV} \\ \dot{NOR} \\ \dot{ALT} \end{bmatrix}^{T}_{1x6}$$

where

$$\hat{AV} \equiv \frac{\vec{R} \times \vec{v}}{|\vec{R} \times \vec{v}|} \times \frac{\vec{R}}{|\vec{R}|} = \text{coordinate along velocity}$$

$$\hat{NOR} \equiv \frac{\vec{R} \times \vec{v}}{|\vec{R} \times \vec{v}|} = \text{coordinate normal to orbit plane}$$

$$\hat{ALT} = \frac{\vec{R}}{|\vec{R}|} = \text{coordinate in direction of altitude}$$

Dots represent corresponding rates.

For INTYPE = 3 or 4

$$PIN = E \begin{bmatrix} X \\ Y \\ Z \\ \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix}_{6x1} \begin{bmatrix} X \\ Y \\ Z \\ \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix}^{T}_{1x6}$$

POUT is the output covariance matrix in cartesian coordinates mean equator 1950.

CONVPI Transformations

INTYPE = 1 Transformation from launch pad tangent plane to 1950.

Required input constants for operation:

LAUNCH PAD COORDINATES

S(80)	Altitude	(KM)
S(81)	Longitude	(DEG)
S(82)	Latitude	(DEG)

LAUNCH AZIMUTH

S(84)	Azimuth	(DEG)

TIME FROM LAUNCH TO INJECTION

S(83)	Time	DAYS HOURS. MIN SEC

The transformation from launch pad azimuth coordinates, (AV, NOR, ALT) to mean equinox of 1950 (X_{50}, Y_{50}, Z_{50}) is obtained as follows. Subroutine TRAC is called with launch time and launch pad coordinates to obtain a set of orthogonal unit vectors. The unit vectors, in equator of date, are: $\hat{U}_D$, unit up vector through the launch pad, $\hat{E}_D$, unit east vector, and $\hat{N}_D$, unit north vector. These unit vectors written in matrix form represent the transformation from equator of date coordinates to launch pad coordinates of date

$$\underset{3\times1}{\vec{X}_{LP}} = \underset{3\times3}{\begin{bmatrix} \hat{N}_D^T \\ \hat{E}_D^T \\ \hat{U}_D^T \end{bmatrix}} \underset{3\times1}{\vec{X}_{DATE}} \qquad (1)$$

The transformation from launch pad coordinates, (N, E, U), to the desired launch azimuth coordinates, (AV, NOR, ALT), is a rotation about the $\hat{U}$ vector by an angle equal to the launch azimuth.

$$\underset{3x1}{\vec{X}_{LAZ}} = \underset{3x3}{\begin{bmatrix} \cos LAZ & \sin LAZ & 0 \\ -\sin LAZ & \cos LAZ & 0 \\ 0 & 0 & 1 \end{bmatrix}} \underset{3x1}{\vec{X}_{LP}} = \underset{3x3}{(ROT)} \underset{3x1}{\vec{X}_{LP}} \qquad (2)$$

substituting for $\vec{X}_{LP}$ from equation (1) yields

$$\underset{3x1}{\vec{X}_{LAZ}} = \underset{3x3}{(ROT)} \underset{3x3}{\begin{bmatrix} \hat{N}_D^T \\ \hat{E}_D^T \\ \hat{U}_D^T \end{bmatrix}} \underset{3x1}{\vec{X}_{DATE}} \qquad (3)$$

Subroutines ROTEQ and NUTAIT are called to obtain transformation matrix (AN) which is the transformation from mean equinox of 1950 to true equator of date.

$$\underset{3x1}{\vec{X}_{DATE}} = \underset{3x3}{(AN)} \underset{3x1}{\vec{X}_{1950}} \qquad (4)$$

Substituting equation (4) into (3) yields

$$\underset{3x1}{\vec{X}_{LAZ}} = \underset{3x3}{(ROT)} \underset{3x3}{\begin{bmatrix} \hat{N}_D^T \\ \hat{E}_D^T \\ \hat{U}_D^T \end{bmatrix}} \underset{3x3}{(AN)} \underset{3x1}{\vec{X}_{1950}} \qquad (5)$$

or

$$\vec{X}_{1950} = \left\{ (AN)^T \begin{bmatrix} \hat{N}_D^T \\ \hat{E}_D^T \\ \hat{U}_D^T \end{bmatrix}^T (ROT)^T \right\} \vec{X}_{LAZ} = (TRANS)\ \vec{X}_{LAZ} \qquad (6)$$

3x1 3x3 3x3 3x3 3x1 3x3 3x1

The corresponding velocity transformation is the following:

$$\dot{\vec{X}}_{1950} = (TRANS)\ \dot{\vec{X}}_{LAZ} \qquad (7)$$

under the assumption that $(\dot{AN}) = 0$ which implies $(\dot{TRANS}) = 0$.

The 3x3 matrix, (TRANS), is the desired transformation from launch azimuth coordinates to mean equator of 1950. Subroutine PTRAN is called with a 2 in the call list to perform the desired covariance matrix transformation.

$$E\left\{\vec{X}_{1950}\vec{X}^T_{1950}\right\} = \begin{Bmatrix} \underset{3x3}{(TRANS)} & \underset{3x3}{0} \\ \underset{3x3}{0} & \underset{3x3}{(TRANS)} \end{Bmatrix} E\left\{\vec{X}_{LAZ}\vec{X}^T_{LAZ}\right\} \begin{Bmatrix} (TRANS) & 0 \\ 0 & (TRANS) \end{Bmatrix}^T \qquad (8)$$

6x6 6x6 6x6 6x6

where $\vec{X}_{1950}$ and $\vec{X}_{LAZ}$ now represent the total state vector (position and velocity).

<u>INTYPE</u> = 2 Transformation from injection tangent plane to 1950.

The transformation from injection tangent plane coordinates, (AV, NOR, ALT), to mean equinox 1950 is obtained as follows. The unit vectors for the injection tangent plane coordinates are obtained in equator of 1950.

$$AV_{50} = \frac{\vec{R} \times \vec{v}}{|\vec{R} \times \vec{v}|} \times \frac{\vec{R}}{|\vec{R}|}$$

$$NOR_{50} = \frac{\vec{R} \times \vec{v}}{|\vec{R} \times \vec{v}|} \tag{9}$$

$$ALT_{50} = \frac{\vec{R}}{|\vec{R}|}$$

These vectors written in matrix form represent the transformation from mean equinox of 1950 to injection tangent plane

$$\underset{3x1}{\vec{X}_{ITP}} = \underset{3x3}{\begin{bmatrix} \hat{AV}_{50}{}^{T} \\ \hat{NOR}_{50}{}^{T} \\ \hat{ALT}_{50}{}^{T} \end{bmatrix}} \quad \underset{3x1}{\vec{X}_{1950}} \tag{10}$$

or

$$\underset{3x1}{\vec{X}_{1950}} = \underset{3x3}{\begin{bmatrix} \hat{AV}_{50}{}^{T} \\ \hat{NOR}_{50}{}^{T} \\ \hat{ALT}_{50}{}^{T} \end{bmatrix}^{T}} \quad \underset{3x1}{\vec{X}_{ITP}} = \underset{3x3}{(TRANS)} \ \underset{3x1}{\vec{X}_{ITP}} \tag{11}$$

The corresponding velocity transformation is

$$\underset{3x1}{\dot{\vec{X}}_{1950}} = \underset{3x3}{(TRANS)} \ \underset{3x1}{\dot{\vec{X}}_{ITP}} \tag{12}$$

Subroutine PTRAN is called with a 2 in the call list and the operation presented in equation (8) is performed with the 3x3 matrix, (TRANS), defined in equation (11).

INTYPE = 3 Transformation from equator of date to equator of 1950.

The transformation from equator of date to equator of 1950 is obtained as follows. Subroutines ROTEQ and NUTAIT are called to obtain (AN), the required transformation matrix.

$$\underset{3x1}{\vec{X}_{DATE}} = \underset{3x3}{(AN)} \underset{3x1}{\vec{X}_{1950}} \tag{13}$$

or

$$\underset{3x1}{\vec{X}_{1950}} = \underset{3x3}{(AN)^T} \underset{3x1}{\vec{X}_{DATE}} = \underset{3x3}{(TRANS)} \underset{3x1}{\vec{X}_{DATE}} \tag{14}$$

The corresponding velocity transformation is

$$\dot{\vec{X}}_{1950} = (TRANS) \dot{\vec{X}}_{DATE} \tag{15}$$

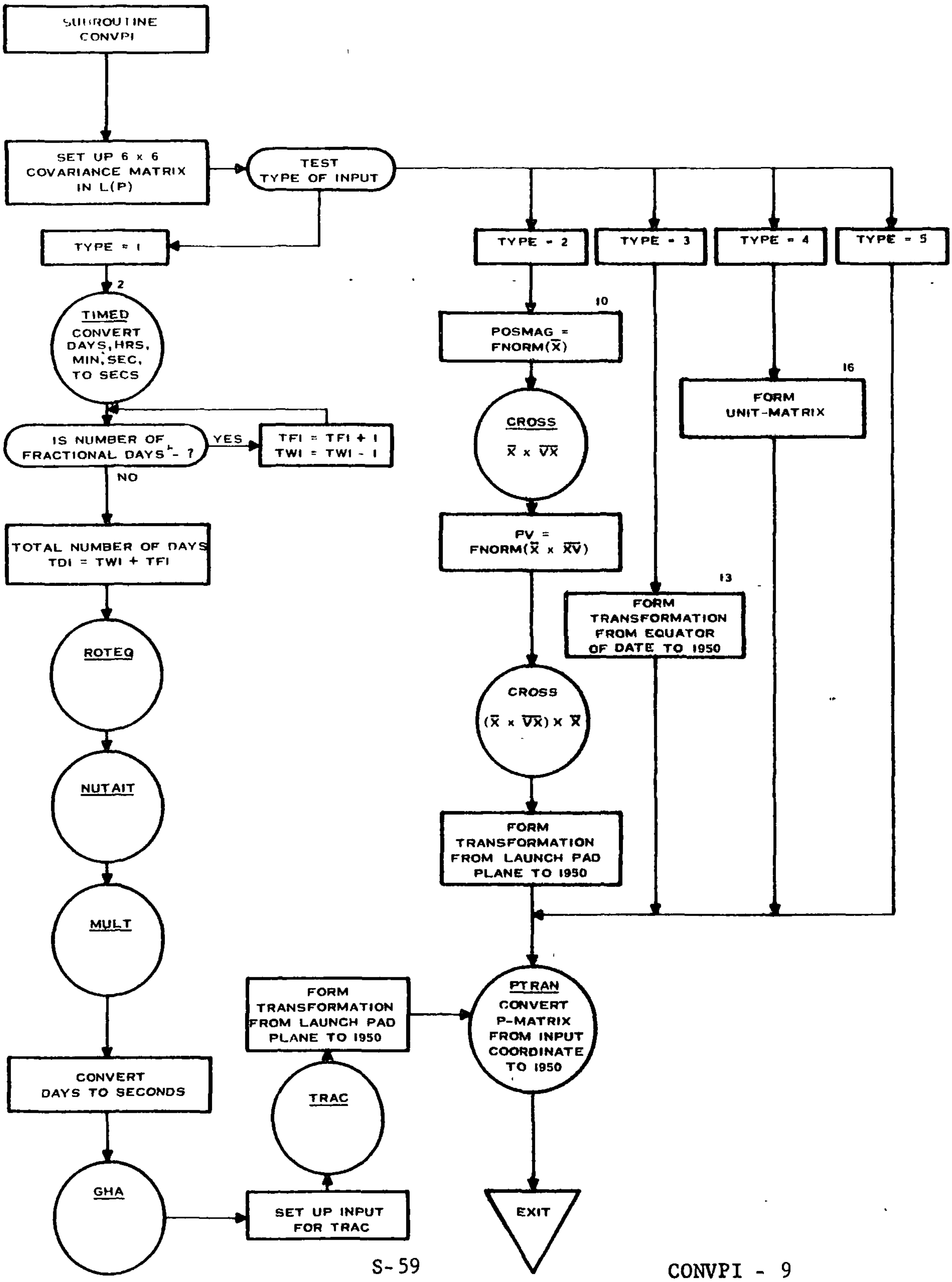
Subroutine PTRAN is called with a 2 in the call list and the operation presented in equation (8) is performed with the 3x3 matrix, (TRANS), defined in equation (14).

INTYPE = 4 Transformation from equator of 1950 to equator of 1950.

The transformation is the unit matrix.

$$\underset{3x1}{\vec{X}_{1950}} = \underset{3x3}{(I)} \underset{3x1}{\vec{X}_{1950}} = (TRANS) \vec{X}_{1950} \tag{16}$$

INTYPE = 5 The subroutine calls PTRAN to perform the operation presented in equation (8) with the transformation matrix, (TRANS), which is in storage from the previous call of CONVPI.


SUBROUTINE CONVPI
SET UP 6 x 6 COVARIANCE MATRIX IN L(P)
TEST TYPE OF INPUT
TYPE = 1
2
TIMED
CONVERT DAYS, HRS, MIN, SEC, TO SECS
IS NUMBER OF FRACTIONAL DAYS - ?
YES
TFI = TFI + 1
TWI = TWI - 1
NO
TOTAL NUMBER OF DAYS TDI = TWI + TFI
ROTEQ
NUTAIT
MULT
CONVERT DAYS TO SECONDS
GHA
SET UP INPUT FOR TRAC
TRAC
FORM TRANSFORMATION FROM LAUNCH PAD PLANE TO 1950
TYPE = 2
TYPE = 3
TYPE = 4
TYPE = 5
10
POSMAG = FNORM($\overline{X}$)
CROSS
$\overline{X}$ x $\overline{VX}$
PV = FNORM($\overline{X}$ x $\overline{XV}$)
CROSS
($\overline{X}$ x $\overline{VX}$) x $\overline{X}$
FORM TRANSFORMATION FROM LAUNCH PAD PLANE TO 1950
13
FORM TRANSFORMATION FROM EQUATOR OF DATE TO 1950
16
FORM UNIT-MATRIX
PTRAN
CONVERT P-MATRIX FROM INPUT COORDINATE TO 1950
EXIT

```
*       LABEL                                                              CONV
*       SYMBOL TABLE                                                       CONV
CEC2010  SUBROUTINE CONVPI                                                 CONV
        SUBROUTINE CONVPI(INPUTP,PI,PS)                                    CONV0030
        COMMON T,S,C,IC                                                    CONV0040
        DIMENSION T(1360),S(1000),C(1000),IC(1)                            CONV0050
       1,PI(21),P(6,6),A(3,3),EM(3,3),AN(3,3),U(3)                         CONV0060
       2,E(3),EN(3),PDUMP(3,3),DUM(3,3),PS(6,6)                            CONV0070
       3,X(3),VX(3), ORORB(3),ALV(3),RT(3)                                 CONV0080
        EQUIVALENCE (T,TDUM),(S,SDUM),(C,CDUM),(IC,ICDUM)                  CONV0090
       1,(S(70),DTS),(S(72),DR),(S(83),TLTI)                               CONV0100
       2,(C(138),AN), (S(115),X),(S(118),VX),(C(10),TW)                    CONV0110
       3,(C(11),TF),(C(129),A),(C(120),EM)                                 CONV0120
C       INPUTP DETERMINES TYPE OF INPUT 1.LAUNCH PAD TANGENT PLANE         CONV0130
C       2.INJECTION TANGENT PLANE 3.EQUATOR DATE 4.EQ1950 5.USE MATRIX     CONV0140
C       THE REQUIRED ORDER OF INPUT VECTOR OF WHICH PI IS THE COVARIANCE   CONV0150
C       MATRIX FOR TANGENT PLANE CASES IS FOLLOWING:                       CONV0160
C       1.ALONG VELOCITY,2.NORMAL TO ORBIT,3.ALTITUDE,4,5,6 CORRES.RATES   CONV0170
        K=0                                                                CONV0180
        DO 1 I=1,6                                                         CONV0190
        I=I                                                                CONV0200
        DO 1 J=I,6                                                         CONV0210
        K=K+1                                                              CONV0220
        P(I,J)=PI(K)                                                       CONV0230
        P(J,I)=P(1,J)                                                      CONV0240
    1   CONTINUE                                                           CONV0250
        INPUTP=INPUTP                                                      CONV0260
        GO TO (2,10,13,16,8),INPUTP                                        CONV0270
    2   CONTINUE                                                           CONV0280
        CALL TIMED(TLTI,TT)                                                CONV0290
        TFI=TF-(TT/86400.)                                                 CONV0300
        TWI=TW                                                             CONV0310
    3   CONTINUE                                                           CONV0320
        IF(TFI)4,5,5                                                       CONV0330
    4   CONTINUE                                                           CONV0340
        TFI=TFI+1.                                                         CONV0350
        TWI=TWI-1.                                                         CONV0360
        GO TO 3                                                            CONV0370
    5   CONTINUE                                                           CONV0380
        TDI=TWI+TFI                                                        CONV0390
        CALL ROTEQ(TDI,A)                                                  CONV0400
        CALL NUTAIT(TDI,OM,CR,DDC,EM,EPSIL)                                CONV0410
        CALL MULT(EM,A,AN,N)                                               CONV0420
        FSEC=DTS*TFI                                                       CONV0430
        CALL GHA(FSEC,TWI,GH,EM(2,1),OMEGA)                                CONV0440
        GH=GH*DR                                                           CONV0450
        AT=S(80)*DR                                                        CONV0460
        ON=S(81)*DR                                                        CONV0470
        AL=S(82)                                                           CONV0480
        CALL TRAC(ON,AT,AL,GH,U,E,EN,RT,AC,SL,CL,ST,CT)                    CONV0490
        FAZ=S(84)*DR                                                       CONV0500
```

```
      SAZ=SINF(FAZ)                                                     CONV05
      CAZ=COSF(FAZ)                                                     CONV05
      DO 6 I=1,3                                                        CONV05
      PDUMP(I,1)=CAZ*EN(I)+SAZ*E(I)                                     CONV05
      PDUMP(I,2)=-SAZ*EN(I)+CAZ*E(I)                                    CONV05
      PDUMP(I,3)=U(I)                                                   CONV05
    6 CONTINUE                                                          CONV05
      DO 7 I=1,3                                                        CONV05
      DO 7 J=1,3                                                        CONV05
      DUM(I,J)=0.                                                       CONV06
      DO 7 K=1,3                                                        CONV06
      DUM(I,J)=DUM(I,J)+AN(K,I)*PDUMP(K,J)                              CONV06
    7 CONTINUE                                                          CONV06
C     DUM IS TRANSFORMATION FROM LAUNCH PAD PLANE TO 1950               CONV06
    8 CONTINUE                                                          CONV06
      CALL PTRAN(P,DUM,PS,2)                                            CONV06
C     PTRAN CONVERTS P MATRIX FROM INPUT COORDINATES TO EQ 1950         CONV06
      GO TO 15                                                          CONV06
   10 CONTINUE                                                          CONV06
      POSMAG=FNORM(X)                                                   CONV07
      CALL CROSS(X,VX, ORORB)                                           CONV07
      PV=FNORM(ORORB)                                                   CONV07
      CALL CROSS (ORORB,X,ALV)                                          CONV07
      AL=POSMAG*PV                                                      CONV07
      DO 11 I=1,3                                                       CONV07
      PDUMP(I,1)=ALV(I)/AL                                              CONV07
      PDUMP(I,2)= ORORB(I)/PV                                           CONV07
      PDUMP(I,3)=X(I)/POSMAG                                            CONV07
   11 CONTINUE                                                          CONV07
      DO 12 I=1,3                                                       CONV08
      DO 12 J=1,3                                                       CONV08
      DUM(I,J)=0.                                                       CONV08
      DO 12 K=1,3                                                       CONV08
      DUM(I,J)=DUM(I,J)+AN (K,I)*PDUMP(K,J)                             CONV08
   12 CONTINUE                                                          CONV08
C     DUM IS TRANSFORMATION FROM INJECTION TAN.PLANE TO 1950            CONV08
      GO TO 8                                                           CONV08
   13 CONTINUE                                                          CONV08
      DO 14 I=1,3                                                       CONV08
      DO 14 J=1,3                                                       CONV09
      DUM(I,J)=AN (J,I)                                                 CONV09
C     DUM IS TRANSFORMATION FROM EQUATOR OF DATE TO 1950                CONV09
   14 CONTINUE                                                          CONV09
      GO TO 8                                                           CONV09
   16 CONTINUE                                                          CONV09
      DO 17 I=1,3                                                       CONV09
      DO 18 J=I,3                                                       CONV09
      DUM(I,J)=0.                                                       CONV09
      DUM(J,I)=DUM(I,J)                                                 CONV09
   18 CONTINUE                                                          CONV10
      DUM(I,I)=1.                                                       CONV10
```

```
17  CONTINUE                                                        CONV1020
    GO TO 8                                                         CONV1030
15  CONTINUE                                                        CONV1040
    RETURN                                                          CONV1050
    END                                                             CONV
```

Subroutine: CORRTP

Purpose: To perform the filtering or weighting of the observation being made and updating of the state covariance matrix for the inclusion of the observation.

Calling Sequence:

CALL CORRTP (QQ, H, P)

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	QQ	1	σ^2_{MEAS}		Measurement variance
I	H	(1,6)	$\frac{\partial \text{ MEAS}}{\partial \text{ STATE}}$		Measurement partials
I,O	P	(6,6)	$E(\vec{x}\,\vec{x}^T)$		State covariance matrix

Common storages used or required: None

Subroutines required: None

Functions required: None

Approximate number of storages required: 124 DEC

Equation Being Solved

The updating of the state covariance matrix for an observation assuming the Schmidt-Kahlman optimum filter is used on the data is the following:

$$P_{NEW} = P_{OLD} - P_o H^T (H P_o H^T + QQ)^{-1} H P_o$$

```
*       LABEL                                                    CORR
*       SYMBOL TABLE                                             CORR
CEC2025  SUBROUTINE CORRTP                                       CORR
        SUBROUTINE CORRTP(QQ,H,P)                                CORR00
        DIMENSION P(6,6),H(6),PMT(6)                             CORR00
        DIMENSION PCOR(6,6)                                      CORR00
        BB=0.                                                    CORR00
        DO 43 J=1,6                                              CORR00
        PMT(J)=0.                                                CORR00
        DO 44 K=1,6                                              CORR00
        PMT(J)=PMT(J)+P(J,K)*H(K)                                CORR00
   44   CONTINUE                                                 CORR00
        BB=BB+PMT(J)*H(J)                                        CORR00
   43   CONTINUE                                                 CORR01
        BB = BB+QQ                                               CORR01
        DO 45 J=1,6                                              CORR01
        DO 45 K=1,6                                              CORR01
        PCOR(J,K)=(PMT(J)*PMT(K))/BB                             CORR01
        P(J,K)=P(J,K)-PCOR(J,K)                                  CORR01
   45   CONTINUE                                                 CORR01
        RETURN                                                   CORR01
        END                                                      CORR
```

Page Intentionally Left Blank

Subroutine: CROSS

Purpose: To find the cross product of two 3-dimensional vectors.

Calling Sequence:

CALL CROSS (A, B, C)

Input and **Output**

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	A	3			First input vector
I	B	3			Second input vector
O	C	3			Result = AXB

Common storages used or required: None

Subroutines required: None

Functions required: None

Approximate number of storages required:

```
*     LABEL                                              CROS
*     SYMBOL TABLE                                       CROS
      SUBROUTINE CROSS(A,B,C)                            CROS0000
      DIMENSION A(3),B(3),C(3)                           CROS0010
      C(1)=A(2)*B(3)-A(3)*B(2)                           CROS0020
      C(2)=A(3)*B(1)-A(1)*B(3)                           CROS0030
      C(3)=A(1)*B(2)-A(2)*B(1)                           CROS0040
      RETURN                                             CROS0050
      END                                                CROS
```

Subroutine: (CSH)S

Purpose: This is the standard Fortran II card image input subroutine modified to output the card image, accept logical input and output tape numbers from SETN, and call subroutine HOUR to read the clock if desired.

Calling Sequence:

CALL (CSH)S (BUFF)

Input and **Output**

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
O	BUFF	12			Buffer for contents of card read

Common storages used or required: None.

Subroutines required: SETN, HOUR, System read routines.

Functions required: None.

Approximate number of storages required:

Page Intentionally Left Blank

Subroutine: DE6FN

Identification

RW DE6FN
704 - FORTRAN SAP Language Subroutine
STL

Purpose

This FØRTRAN subprogram integrates a set of N simultaneous second-order ordinary differential equations in which first derivatives may or may not appear. It is the FØRTRAN version of the standard subroutine RW DE6F.

Restrictions

Same as explained in the write-up for RW DE6F.

Method

A fourth-order Runge-Kutta method (RW DE5F) is used to start the integration and to change the step-size during integration. A Cowell "second-sum" method based on sixth differences is used to continue the integration. While input to this routine is single precision, double precision is used internally to control round-off errors. Truncation error can be controlled by choosing an appropriate step-size, or by using the variable step-size mode of operation. The set-up entry uses the auxiliary subroutine to evaluate the second-order derivatives. The values of the variables and derivatives are consistent at all times. A detailed description of the method used is available in Appendix A.

Usage

a. Calling LIST for set-up entry (performed prior to initiating the integration).

```
CALL DE6FN(I0,I1,T(1),N,V,J,K,A3,A4,A5,A6)
```

I0 is the same as P0 of RW DE6F write-up.

- +1 1st derivatives are present in the evaluation of the second derivatives.
- -1 1st derivatives are missing in the evaluation of the second derivatives.

I1 is the same as P1 of RW DE6F write-up.

- +1 Variable step-size mode of operation is used.
- -1 Fixed step-size mode of operation is used.

T is the name of the floating point array which is of dimension 30N+3 and is reserved by the user. This region should be located in COMMON, since it must be referred to by the main program and by the subroutine V. T(1) need not be set equal to N; however, all other requirements that concern the usage of area T apply. The value of N is not available to the programmer in T(1) and if it is required by the auxiliary subroutine, it should appear in COMMON also.

N is the number of equations (an integer).

V is the name of a FØRTRAN subroutine for evaluating the derivatives y_i. This subroutine must be named in the main program by the use of an "F" card.

J equates to B in RW DE6F write-up (an integer).

K equates to R in RW DE6F write-up (an integer).

A3 equates to $\alpha+3$ in RW DE6F write-up (floating point).

A4 equates to $\alpha+4$ in RW DE6F write-up (floating point).

A5 equates to $\alpha+5$ in RW DE6F write-up (floating point).

A6 equates to $\alpha+6$ in RW DE6F write-up (floating point).

For meaning of N,V,J,K,A3,A4,A5,A6 refer to RW DE6F write-up.

Region T contains the following information prior to Set-up Entry.

T(2)	x	value of independent variable
T(3)	h	value of step-size
T(4) thru	y_i thru	values of the independent variables y_i
T(N+3)	y_N	
T(N+4) thru	y_i' thru	values of the first derivatives y_i'
T(2N+3)	y_N'	
T(2N+4) thru	y_i'' thru	values of the second derivatives y_i''
T(3N+3)	y_N''	
T(3N+4) - T(30N+4)		temporary storage

More detailed description of this storage may be found in the RW DE6F write-up.

b. RW DE6FN entry to integrate one step.

CALL DE6FPI (ACCUM)

Upon return ACCUM will have the results which formerly appeared in the accumulator.

c. RW DE6FN entry for a running start.

CALL DE6FP2

d. RW DE6FN entry for a change of step-size in Cowell/Runge-Kutta System.

CALL DE6FP3(H)
H is the new value of h which was formerly placed in the accumulator prior to entry.

e. RW DE6FN entry for a change of step-size for a final integration.

CALL DE6FP4
T(3) must be set to the correct value before the CALL.

f. RW DE6FN entry to initiate a running change in coordinates and to set a list call to a non-zero value.

CALL DE6FNZ (VALUE)

g. RW DE6FN entry to change from variable mode to fixed mode.

CALL DE6FNG

h. RW DE6FN entry to change from fixed mode to variable mode.

CALL DE6FPS

i. RW DE6FN entry to change the value of h_{min}.

CALL DE6FMN (VALUE)
VALUE is the new h_{min} in floating point.

j. RW DE6FN entry to change the value of h_{max}.

CALL DE6FMA (VALUE)
VALUE is the new h_{max} value in floating point.

k. RW DE6FN entry to change the value of y_{min}.

CALL DE6FCM (VALUE)
VALUE is the new y_{min} value in floating point.

l. RW DE6FN entry to change the value of R after a Cowell integration step.

CALL DE6FCH(NR,R)
NR is the integer value of R.
R is the value in floating point.

Identification

RWDE6F Floating Point Cowell (Second Sum), Runge-Kutta
Integration of Second-Order Equations
704 - SAP
STL

Purpose

To solve a set of N simultaneous second-order ordinary differential equations in which first derivatives may or may not appear.

Restrictions

No internal checks are made for overflow or underflow. The user must provide an auxiliary subroutine which evaluates the second-order derivatives. The initial conditions must be set up prior to the first set up entry.

Method

A fourth-order Runge-Kutta method* (RWDE5F) is used to start the integration and to change the step-size during integration. A Cowell "second-sum" method based on sixth differences is used to continue the integration. While input to this routine is single precision, double precision is used internally to control round-off errors. Truncation error can be controlled by choosing an appropriate step-size, or by using the variable step-size mode of operation. The set-up entry uses the auxiliary subroutine to evaluate the second-order derivatives. The values of the variables and derivatives are consistent at all times. A detailed description of the method used is available in Appendix A of this Subroutine.

Usage

Calling sequence to set up a problem:

Loc.	Instruction	Comments
γ	TSX DE6F,4	Set up entry.
$\alpha+1$	PO T,O,V	Option, addresses of storage and auxiliary subroutines.
$\alpha+2$	P1 B,O,R	Option and Parameters
$\alpha+3$	DEC 1E-S	S is the number of significant figures desired.
$\alpha+4$	DEC h_{min}	Minimum step-size. Floating point.

* Scarborough, T.B., Numerical Mathematical Analysis, Third Edition, John Hopkins Press, Baltimore, 1955 (pp. 301-302)

Loc.	Instruction	Comments
$\alpha+5$	DEC h_{max}	Maximum step-size. Floating point.
$\alpha+6$	DEC y_{min}	Minimum y_i value allowed for testing. Floating point. (See Appendix A for details of y_{min}.)

Calling sequence to integrate all variables one step:

α	TSX DE6F+1,4	Integration entry.
$\alpha+1$	Return	Sign of AC will be plus if the integration was Runge-Kutta and minus if Cowell.

The address T is the first of 30N+3 calls arranged as follows:

T	PZE N,0,0	N is the number of equations. Fixed point.
T+1	x	Value of independent variable. Floating point.
T+2	h	Value of step-size. Floating point.
T+23 thru T+N+2	y_i thru y_N	Values of the dependent variables y_i. Floating point
T+N+3 thru T+2N+2	y'_i thru y'_N	Values of the first derivatives y'_i Floating point
T+3N+3 thru T+3N+3	y''_i thru Y''_N	Locations where the user's auxiliary subroutine must store the second derivatives y''_i .
T+3N+3 thru T+30N+2		27N cells of temporary storage.

T+3N+3 thru T+9N+2 (6N) are used by the Runge-Kutta subroutine (RWDE5F). T+4N+3 thru T+5N+2 and T+8N+3 thru T+9N+2 contain the least significant parts (except when a change of coordinates is in progress) of y'_i and y_i respectively, and must be preserved throught the entire integration procedure. The final 21N cells of the T storage are used by the Cowell subroutine and must also be preserved. (See Appendix E of this subroutine for a detailed description of these 27N cells.)

The address V is the entry point of the auxiliary subroutine which evaluates the derivatives y''_i and is entered by the calling sequence:

α	TSX V,4	Index registers need not be saved in V.
$\alpha+1$	Return	Return must be made via a TRA 1,4.

The first B ($\leq$ N) equations are tested to determine whether it is necessary to halve or possibly to double the step-size or to proceed with a Cowell integration step.

For a given step-size h, the Cowell integration step is h and the Runge-Kutta integration step is h/R.

Options

PO - PZE 1st derivatives are present in the evaluation of the second derivatives
- MZE 1st derivatives are missing in the evaluation of the second derivatives.

P1 = PZE Variable step-size mode of operation is used.
= MZE Fixed step-size mode of operation is used.

If 1E-S, h_{min}, h_{max}, and y_{min} are not specified (0 in first calling sequence), the subroutine will set them to 1E-9, 0, 1E18, and 1., respectively.

Special Usage (See Appendices A and B for complete details.)

The following special usages are possible:

1. Running change of coordinates.
2. Running start.
3. Change of step-size by user in the Cowell/Runge-Kutta system.
4. Change of step-size for a final integration or at some prescribed value of x.

Space Required (In addition to T and V).

955 calls of program and constants. (Includes DE5F.)

44 calls of COMMON thru COMMON + 43.

Timing

Set-up time. (V=time for 1 entry to the auxiliary subroutine.)

$.012 \left[12N + 512\right]$ ms. + 1V.

To integrate one step:

1. Runge-Kutta (AC=+ after an integration return.)

A. With 1st derivatives.

$.012 \left[476N + (8N+18)/R + 240\right]$ ms. + 4V.

B. Without 1st derivatives.

$.012 \left[383N + (8N + 18)/R + 240\right]$ ms. + 3V.

Timing (Continued)

2. Cowell (AC= - after an integration return.)

A. With 1st derivatives.

$$.012 \left[2901N + 92B + 194\right] \text{ ms.} + 2V.$$

$$+ .012 \left[2124N + 34\right] \text{ ms. if previous integration was Runge-Kutta.}$$

B. Without 1st derivatives.

$$.012 \quad 2334N + 92B + 194 \quad \text{ms.} + 2V.$$

$$+ .012 \quad 2124N + 88B + 34 \quad \text{ms. if previous integration was Runge-Kutta.}$$

3. Change of Coordinates. (In addition to first part of 2A or 2B.)

$$.012 \left[2286N\right] \text{ ms.} + 8V.$$

APPENDIX A

METHOD

This routine is prepared to solve the following system:

$$1) \quad \left.\begin{array}{l} y_i'' = f_i\,(x, y_i, \ldots, y_N,\ y_i', \ldots, y_N') \\[2ex] y_i(x_0) = y_{i0},\quad y_i(x_0) = y'_{i0} \end{array}\right\} \quad i = 1,\ 2,\ \ldots,\ N$$

In case none of the f_i involve the first derivatives y_i', time is saved by indicating this in the set-up. The Runge-Kutta routine RWDE5F is used to start the integration and also to change the step-size h. The user must ask for each integration step, and the routine will follow this sequence:

1. R Runge-Kutta steps of size $\frac{h\ start}{R}$ are taken to obtain y_{i1}, y_{i1}', y_{i1}''. This is continued until we reach y_{i6}, y_{i6}', y_{i6}'', after a total of 6R Runge-Kutta steps. The integer R (=4 if unspecified) simply allows Runge-Kutta to operate at a smaller step than the main program.

2. For each of the N equations, that part of the difference table above the diagonal line is constructed in three steps:

		y''_{i0}						
			Δ^{I}_{i0}					
		y''_{i1}		Δ^{II}_{i0}				
			Δ^{I}_{i1}		Δ^{III}_{i0}			
		y''_{i2}		Δ^{II}_{i1}		Δ^{IV}_{i0}		
			Δ^{I}_{i2}		Δ^{III}_{i1}		Δ^{V}_{i0}	
$\underline{{}''F_{i4}}$		$\underline{y''_{i3}}$		$\underline{\Delta^{II}_{i2}}$		$\underline{\Delta^{IV}_{i1}}$		$\underline{\Delta^{VI}_{i0}}$
	$\underline{{}'F_{i4}}$		$\underline{\Delta^{I}_{i3}}$		$\underline{\Delta^{III}_{i2}}$		$\underline{\Delta^{V}_{i1}}$	
$''F_{i5}$		y''_{i4}		Δ^{II}_{i3}		Δ^{IV}_{i2}		Δ^{VI}_{i1}
	$'F_{i5}$		Δ^{I}_{i4}		Δ^{III}_{i3}		Δ^{V}_{i2}	
$''F_{i6}$		y''_{i5}		Δ^{II}_{i4}		Δ^{IV}_{i3}		
	$'F_{i6}$		Δ^{I}_{i5}		Δ^{III}_{i4}			
$''F_{i7}$		y''_{i6}		Δ^{II}_{i5}				
	$'F_{i7}$		Δ^{I}_{i6}					
$''F_{i8}$		y''_{i7}						
	$'F_{i8}$							
$''F_{i9}$								

First the known y''_{i0} through y''_{i6} are differenced to give the right half of the table. Next are calculated in extra precision:

2) $$''F_{i4} = \frac{y_{i3}}{h^2} - C_0\, y''_{i3} - C_2\, \Delta^{II}_{i2} - C_4\, \Delta^{IV}_{i1} - C_6\, \Delta^{VI}_{i0}$$

$$3)\quad {}'F_{i4} = \frac{y'_{i3}}{h} - D_0\, y''_{i3} - D_1\, \Delta^{I}_{i3} - D_2\, \Delta^{II}_{i2} - D_3\, \Delta^{III}_{i2}$$

$$- D_4\, \Delta^{IV}_{i1} - D_5\, \Delta^{V}_{i1} - D_6\, \Delta^{VI}_{i0}$$

The table is then completed down to the diagonal line, by requiring the difference between any entry and the entry above to equal the entry to the right. The constants used in equations (2) - (7) are given in the description of the Livermore Cowell routine.

3. Before going to a Cowell step, the step-size h is tested. The tests are omitted, however, if the user so indicates in the initial calling sequence, in which case h is fixed. Only the first B equations are used to test, where $1 \le B \le N$ and $B = N$ if unspecified. We determine--

$$V = \frac{\max}{1 \le i \le B} \left| \frac{\Delta^{V}_{i1}}{\max(y_{i6},\, y_{min})} \right| . \quad \text{If} \quad V \ge \frac{10^{3-S}}{h^2} \text{ , then the}$$

ratio of 5th difference to function is too large -- if S decimal places are to be retained at that step. Therefore, h is reduced to h/2 and Runge-Kutta re-entered for another sequence of 6R steps. These begin with the latest calculated values (y_{i6}, y'_{i6}) and no ground is retraced. The constant y_{min} (=1 if unspecified) prevents division by y near a zero; for example, in sine calculation y_{min}=.1 avoids difficulty near 180^{o}. The integer S, taken as 9 if unspecified, allows a large h if chosen smaller, say S = 7.

$$\text{If} \quad \frac{10^{-1-S}}{h^2} < V < \frac{10^{3-S}}{h^2} \text{ , we proceed to a Cowell step.}$$

If $V \le \dfrac{10^{-1-S}}{h^2}$, we may be able to double h. We test further to see that

$$W = \max_{1 \le i \le B} \left| \frac{\Delta^{VI}_{i0}}{\max(y_{i6}, y_{min})} \right| \le \frac{10^{-1-S}}{h^2},$$

and if so, we re-enter Runge-Kutta after replacing h by 2h, since the step-size h has led to needlessly small difference to function ratios. Of course, h is not halved or doubled if this would violate h_{min} or h_{max}, which are 0 and 10^{18} if unspecified.

4. The Cowell integration begins with predictions:

$$4) \quad y_{i7} = h^2 \left({}''F_{i8} + N_0 y''_{i6} + N_1 \Delta^{I}_{i5} + \ldots + N_6 \Delta^{VI}_{i0} \right)$$

$$5) \quad y'_{i7} = h \left({}'F_{i7} + \dot{N}_0 y''_{i6} + \dot{N}_1 \Delta^{I}_{i5} + \ldots + \dot{N}_6 \Delta^{VI}_{i0} \right)$$

These use the row of the difference table above the diagonal line; only this row is needed for a Cowell step and is kept up to date as the integration proceeds. (We mention that the above prediction for y'_{i7} is omitted in the missing first derivative option.) Now from y_{i7} and y'_{i7}, we obtain y''_{i7} and then complete the row of differences out to Δ^{VI}_{i1} under the diagonal line in the table. With this row, we calculate final corrected values --

$$6) \quad y_{i7} = h^2 \left({}''F_{i8} + B_0 y''_{i7} + \ldots + B_6 \Delta^{VI}_{i1} \right)$$

$$7) \quad y'_{i7} = h \left({}'F_{i7} + \dot{B}_0 y''_{i7} + \ldots + \dot{B}_6 \Delta^{VI}_{i1} \right)$$

From these we get corrected values for y''_{i7}, and recalculate the entire row under the diagonal line. This completes the integration step. Using the new row of differences, the next step begins by testing the step-size (i.e., at 3.).

Further Properties of the Program

In some problems, information about the first derivative (velocity) may be less reliable than information about the function (position). The user may then choose a "running change of coordinates" or a "running start;" these depend on the fact that with 8 consecutive values of the y_i (and the y_i' in case first derivatives are present in the f_i) the Cowell part of the program can be self-starting. The mathematics is simple: step No. 1 is omitted, and No. 2 modified to calculate $''F_{i4}$ and $''F_{i5}$ from Eq. (2) (instead of $''F_{i4}$ and $'F_{i4}$). The difference table may again be completed, and Cowell integration begins. The user, having tested the AC to establish that the previous step was a Cowell step, begins a running change of coordinates by setting cell DE6F + 500 to non-zero. He then sets up a counter and begins immediately to store 8 consecutive values of the y_i starting at T+3+11N (and y_i' starting at T+3+3N, if they appear in the f_i). After changing the coordinates the 8th time, the user may change the second derivative evaluation routines; if x and h are to be in new units this should also be done. When another step is asked for, the routine will form a difference table in the new coordinates and proceed to a Cowell step.

The mechanics of a running start are similar; after going through the set-up routine, the user loads his values of y_{i0} through y_{i7} (and the y_i' if needed in the f_i) into the same locations as above and makes the required transfer.

There may also be occasions on which the user will wish to modify h himself; e.g., if he wishes to produce the numerical solution at some prescribed value of x, or if he wishes to approach a running change of coordinates at a step-size smaller than that being used by the routine. The technique of modifying h is described in Appendix B.

APPENDIX B

USAGE AND CODING INFORMATION

There are essentially two entries to the subroutine. The first entry must be made once at the beginning to set up the addresses, options, parameters, etc., of the routine for integration of N simultaneous second-order, ordinary differential equations, in which first derivatives may or may not appear. The first entry utilizes the auxiliary subroutine to evaluate the second-order derivatives at the initial conditions. Thus, the initial conditions must be set up prior to the first entry. The second entry may be used any number of times after the first to integrate all y_i from x to x+h by a Cowell step; or x to x+h/R by a Runge-Kutta step.

The first entry has the following calling sequence:

Loc.	Instruction	Comments
α	TSX DE6F,4	Set-up entry.
α+1	PO T,O,V	Option, addresses of storage and auxiliary subroutine.
α+2	P1 B,O,R	Option and parameters.
α+3	DEC 1E-S	S is the number of significant figures desired.
α+4	DEC h_{min}	Minimum step-size. Floating point.
α+5	DEC h_{max}	Maximum step-size. Floating point.
α+6	DEC y_{min}	Minimum y_i value allowed for testing. Floating point.
α+7	Return	

(α+1): T is the address of the first word of a block of 30N+3 cells of temporary storage arranged as follows:

Loc.	Contents	Comments
T	PZE N,0,0	N is the number of equations. Fixed point.
T+1	x	Value of independent variable. Floating point.
T+2	h	Value of step-size. Floating point.
T+3 thru T+N+2	y_i thru y_N	Values of the dependent variables y_i. Floating point.
T+N+3 thru T+2N+2	y'_i thru y'_N	Values of the first derivatives y'_i. Floating point.
T+2N+3 thru T+3N+2	y''_i thru y''_N	Locations where the user's auxiliary subroutine must store the second derivatives y''_i. Floating point.
T+3N+3 thru	T+30N+2	27N cells of temporary storage.

The next 27N storages of T are temporary storages. The Runge-Kutta subroutine uses the first 6N cells (T+3N+2 thru T+9N+3) and the Cowell routine uses the final 21N cells (T+9N+3 thru T+30N+2). However, if a change of coordinates (see Special Usage) is made, the Cowell routine will also use the first 6N cells. The attached T Storage Chart shows the set-up of the entire T region. The N cells starting at T+5N+3 and the N cells starting at T+6N+3 are used by the Runge-Kutta subroutine to compute $\Delta' y_{iN}$ and Δy_{iN}. However, $\Delta' y_{iN}$ and Δy_{iN} are destroyed before final exit, and these cells contain intermediate values of no significance to the user. The left side of the chart shows the storage for the normal case where 6R Runge-Kutta steps are taken (using the first 9N cells of T for each integration) before an attempt is made to proceed to a Cowell step. At the beginning of each set of R steps, the Cowell subroutine saves the values of the second derivatives (7 sets starting at T+11N+3). In addition, the values of y_{i3} and y'_{i3} are saved, starting at T+9N+3 and T+10N+3 respectively, for use in the central difference equations where $''F_{i4}$ and $'F_{i4}$ are calculated. Care must be exercised in using certain values in the temporary region.

For instance, after a Runge-Kutta integration step, the most significant values of y_i and y_i', starting at T+7N+3 and T+3N+3 respectively, will be the values of the previous integration; while the least significant values of y_i and y_i', starting at T+8N+3 and T+4N+3, respectively, will be the values of the present integration. The Cowell routine also saves the least significant values of y_i and y_i' (unless a change of coordinates is in progress) in these same storages at the end of each integration. The 11N storages starting at T+19N+3 contain the right half of the N difference tables, an example of which is shown in Appendix A. The right side of the chart shows other values which are stored in the T region during a change of coordinates and will be explained later under Special Usage. Even through only one symbol is given (y_{i0}', etc.), it should be understood that N values are stored as in the left side of the chart. Thus, y_{i0}' signifies y_{10}', y_{20}', y_{30}', ... , y_{NO}'.

The address V is the entry point of an auxiliary subroutine which the user must provide to evaluate the second derivatives y_i''. This subroutine must store y_i'' in T+2N+3 through T+3N+2 as shown above. The subroutine is entered by the calling sequence:

Loc.	Instruction	Comments
α	TSX V,4	Index registers need not be saved.
α+1	Return	Return via a TRA 1,4.

The derivatives y_i'' are evaluated during the set-up and at the end of each integration step. Thus, the values of the variables and the derivatives are consistent at all times. Extra precision is recommended for the evaluation of the second derivatives y_i''.

PO should be set to PZE if the first derivatives are present in the evaluation of the second derivatives. If first derivatives are not present, PO should be set to MZE.

(α+2): P1 should be set to PZE if a variable step-size is wanted. For a fixed step-size; P1 should be set to MZE. The former allows doubling and halving while the latter restricts the routine to a fixed r. The user may change the mode of operation externally at any time by setting cell DE6F+501 to plus for a variable step-size and minus for a fixed step-size.

Only the first B ($1 \leq B \leq N$) equations are tested to determine doubling or halving of h. Thus, the user should arrange the N equations in descending order of importance, and specify B accordingly. If B = 0 in the calling sequence, it will be set to N.

R is the ratio of the Cowell step-size to the Runge-Kutta step-size. Thus, smaller integration steps can be taken in the Runge-Kutta subroutine by setting R greater than 1. If R = 0 in the calling sequence, it will be set to 4. R is saved in the decrement of cell DE6F+516 and in floating point in cell DE6F+517. After any Cowell integration step (AC= -), the user chould change R by changing both of these cells.

(α+3): 1E-S is a floating point number where S is the number of significant figures of accuracy desired at each step. The user should experiment with S to fit his own particular problem. The 1 of 1E-S may also be varied from 1 to 9 (1E-S thru 9E-S) for a final degree of control over the accuracy testing. If 1E-S=0 in the calling sequence, it will be set to 1E-9.

(α+4): h_{min} is a floating point number giving a lower bound for h. h_{min} is saved in cell DE6F+509 and can be changed at any time.

(α+5): h_{max} is a floating point number giving an upper bound for h. If h_{max}=0 in the calling sequence, the upper bound will be set to 1E18. h_{max} is saved in cell DE6F+510 and can be changed at any time.

(α+6): y_{min} is a positive floating point number which is used in testing the step-size. If y_{min}=0 in the calling sequence, it will be set to 1. y_{min} is saved in cell DE6F+511 and can be changed at any time.

If the fixed step-size mode of operation is selected (P1=MZE), then B, 1E-S, h_{min}, h_{max}, and y_{min} are all ignored by the subroutine. (If P1=MZE, set B = 1 for maximum efficiency.)

The integration entry has the following calling sequence:

Loc.	Instruction	Comments
α	TSX DE6F+1,4	Integrates all variables one step.
α+1	Return	

Upon return from the integration entry, the accumulator will be plus if the integration was a Runge-Kutta step and minus if the integration was a Cowell step. Ordinarily, x will have been advanced to x +h/R for a Runge-Kutta step and x + h for a Cowell step. However, in the variable h mode, it is possible that the value of h in T+2 prior to the integration entry has been changed to h/2 or 2h. In this case, the integration step will be a Runge-Kutta step, and the value of x will be either x + h/2R or x + 2h/R. All values of y_i, y_i', and y_i'' will be consistent with the new value of x. The user must never change the value of the step-size h except as described under Special Usage.

Special Usage (See Appendix A. Further Properties of the Program.)

1. Running Change of Coordinates. (Normal Entries.)

After any Cowell integration step (AC= -), the user may initialize the beginning of a change of coordinates by setting cell DE6F+500 to nonzero. Starting with the present values, he begins to save eight consecutive sets of y_i starting at T+11N+3 (and y_i' starting at T+3N+3 if they appear in the y_i''). He continues to use the integration entry above. The routine will detect the non-zero value stored in cell DE6F+500 and will begin a count-down in cell DE6F+502 from 8 (-1) 0. The next seven integrations will be Cowell steps and all testing will be discontinued during this period. After the eighth set of y_i (y_{i7}), and y'_i (y'_{i7}) if necessary, have been stored (DE6F+502 will have a fixed point 1 in the address), the user may change the second derivative evaluation routine V and the units of x and h. The units of the eight sets of y_i and y'_i may be changed while storing each set, or after all eight sets have been stored. When another integration step is asked for, the routine will perform the change of coordinates and proceed to a Cowell integration step. Cell DE6F+500 will be restored to zero, and an 8 will be restored to the address of cell DE6F+502. Thus, the routine will be ready for another change of coordinates and will operate under standard conditions.

2. Running Start. (Special Entry.)

A running start is similar to a running change of coordinates except that the user must supply all eight sets of y_i (and y'_i if necessary) at one time. The following sequence of operations must be followed:

A. Set up the initial conditions in the T storage. Only N, x, and h are needed, although the eight sets of y_i (but not y'_i) can also be stored at this time. x must correspond to y_{i7} and h must be the interval at which the y_i have been obtained. Thus, $x = x_0 + 7h$ where x_0 corresponds to y_0 (and y'_0).

B. Use the first entry calling sequence to set up all parameters and options. The V subroutine will be used but will have no effect on the problem. Also, cells T+4N+3 thru T+5N+2 and T+8N+3 thru T+9N+2 will be set to zero. Thus, the eight sets of y'_i must be stored after the setup entry.

c. Store eight consecutive sets (equal intervals) of y_i starting at cell T+11N+3 (and eight consecutive sets of y'_i starting at cell T+3N+3, if needed).

D. Execute the following calling sequence one time:

Loc.	Instruction	Comments
α	TSX DE6F+16,4	Enter only once.
α+1	Return	Integrates 1 step.

E. Continue with the regular integration entry (TSX DE6F+1,4) to integrate further. Step D integrates all variables one Cowell step, and $x(x_0 + 7h)$ is advanced to $x + h(x_0 + 8h)$. From this point, the routine will operate under normal conditions.

3. Change of Step-Size in the Cowell/Runge-Kutta System. (Special Entry.)

During the integration procedure, the user may wish to output for a specific value of x without interrupting the Cowell/Runge-Kutta integration procedure. Or, he may wish to change the value of the step size h prior to a running change of coordinates. He can do this after any integration entry with the following procedure:

Loc.	Instruction	Comments
α	TSX DE6F+22,4	Changes h and starts new series of 6R Runge-Kutta steps.
α+1	Return	

Thus, the integration step will be h/R. Continue with the regular integration entry (TSX DE6F+1,4) to integrate further.

If the above procedure is being used to reduce the step-size prior to a change of coordinates, the user must prevent the routine from doubling again until after the change of coordinates. Doubling can be prevented either by storing zero in cell DE6F+510 (h_{max}), or by setting cell DE6F+501 (fixed step size) negative prior to the above entry. After the change of coordinates, the user may restore the above cells.

4. Change of Step Size for a Final Integration. (Special Entry.)

This is simply a direct transfer to the Runge-Kutta integration subroutine and should be used to end exactly at a specific value of x.

After changing the value of h in T+2,

Loc.	Instruction	Comment
α	TSX DE6F+588,4	Integrates one step with the Runge-Kutta subroutine.
α+1	Return	

The integration step will be the value of h in T+2. This procedure could be used in the middle of the integration procedure if 3. above (TSX DE6F+22,4) is used immediately afterwards to restart in the Cowell/ Runge-Kutta system.

In addition to the user's auxiliary subroutine and the 30N+3 cells for the T storage, the storage requirements are 955 words for RWDE6F plus 44 words of COMMON.

The value of the independent variable x is accumulated in double precision when incremented by h. The least significant part of x is saved in cell DE6F+718.

APPENDIX B - T STORAGE CHART

	N	x	h					
T+3	y_1	y_2	·	·	y_N	Initial conditions which user must supply.		
T+N+3	y'_1	y'_2	·	·	y'_N			
T+2N+3	y''_1	y''_2	·	·	y''_N	2nd derivatives stored by V subroutine.		
T+3N+3	y'_1	y'_2	·	·	y'_N	Most significant	y'_{10}	y''_{10}
T+4N+3	y'_1	y'_2	·	·	y'_N	Least significant	y'_{11}	y''_{11}
T+5N+3	$\Delta y'_1$	$\Delta y'_2$	·	·	$\Delta y'_N$	Destroyed by R.K.	y'_{12}	y''_{12}
T+6N+3	$\Delta y'_1$	$\Delta y'_2$	·	·	$\Delta y'_N$	Destroyed by R.K.	y'_{k3}	y''_{k3}
T+7N+3	y_1	y_2	·	·	y_N	Most significant	y'_{14}	y''_{14}
T+8N+3	y_1	y_2	·	·	y_N	Least significant	y'_{15}	y''_{15}
T+9N+3	y_{13}	y_{23}	·	·	y_{N3}	Saved for central	y'_{16}	y''_{16}
T+10N+3	y'_{13}	y'_{23}	·	·	y'_{N3}	difference equations	y'_{17}	y''_{17}
T+11N+3	y''_{10}	y''_{20}	·	·	y''_{N0}	<u>Normal Case</u>.	y_{10}	
T+12N+3	y''_{11}	y''_{21}	·	·	y''_{N1}	Saved from 6R R-K	y_{11}	
T+13N+3	y''_{12}	y''_{22}	·	·	y''_{N2}	steps. Used to form	y_{12}	
T+14N+3	y''_{13}	y''_{23}	·	·	y''_{N3}	difference tables in	y_{13}	
T+15N+3	y''_{14}	y''_{24}	·	·	y''_{N4}	an attempt to proceed	y_{14}	
T+16N+3	y''_{15}	y''_{25}	·	·	y''_{N5}	to a Cowell step.	y_{15}	
T+17N+3	y''_{16}	y''_{26}	·	·	y''_{N6}		y_{16}	
T+18N+3	Used only if coordinates changed.						y_{17}	

<u>Coordinate Change</u> (T+3N+3 to T+10N+3)

8 sets of consecutive y'_i saved by user if $y'' = f(x,y,y')$. Cowell stores y''_i over y'_i.

<u>Coordinate Change</u> (T+11N+3 to T+18N+3)

8 sets of consecutive y_i saved by user. Cowell uses (with V sub) to form y''_i above.

T+19N+3	Δ^{VI}_{10}	Δ^{V}_{11}	Δ^{IV}_{12}	Δ^{III}_{13}	Δ^{II}_{14}	Δ^{I}_{15}	y''_{16}	$''F_{18}$	$''F_{18}$	$'F_{17}$	$'F_{17}$
T+19N+14	Δ^{VI}_{20}	Δ^{V}_{21}	Δ^{IV}_{22}	Δ^{III}_{23}	Δ^{II}_{24}	Δ^{I}_{25}	y''_{26}	$''F_{28}$	$''F_{28}$	$'F_{27}$	$'F_{27}$
T+19N+25	Δ^{VI}_{30}	Δ^{V}_{32}	Δ^{IV}_{32}	Δ^{III}_{33}	Δ^{II}_{34}	Δ^{I}_{35}	y''_{36}	$''F_{38}$	$''F_{38}$	$'F_{37}$	$'F_{37}$
·	·	·	·	·	·	·	·	↑	↑	↑	↑
·	·	·	·	·	·	·	·	Most	Lst.	Most	Lst.
·	·	·	·	·	·	·	·	Sig. ↓	Sig. ↓	Sig. ↓	Sig. ↓
T+30N-8	Δ^{VI}_{N0}	Δ^{V}_{N1}	Δ^{IV}_{N2}	Δ^{III}_{N3}	Δ^{II}_{N4}	Δ^{I}_{N5}	y''_{N6}	$'F_{N8}$	$''F_{N8}$	$'F_{N7}$	$'F_{N7}$

11N storages.

N sets of differences saved for next Cowell step.

(See Appendix A)

```
*       FAP                                                          DE6FN
        TTL DE6FN  COWELL INTEGRATION ROUTINE  CORRECTED 6/63        DE6FN0000
        LBL DE6FN                                                    DE6FN0010
        COUNT   1100                                                 DE6FN0020
        ENTRY DE6FN                                                  DE6FN0030
        ENTRY DE6FP1                                                 DE6FN0040
        ENTRY DE6FP2                                                 DE6FN0050
        ENTRY DE6FP3                                                 DE6FN0060
        ENTRY DE6FNZ                                                 DE6FN0070
        ENTRY DE6FPS                                                 DE6FN0080
        ENTRY DE6FNG                                                 DE6FN0090
        ENTRY DE6FCH                                                 DE6FN0100
        ENTRY DE6FCM                                                 DE6FN0110
        ENTRY DE6FP4                                                 DE6FN0120
        ENTRY DE6FMN                                                 DE6FN0130
        ENTRY DE6FMA                                                 DE6FN0140
        PZE 0                                                        DE6FN0150
        PZE 0                                                        DE6FN0160
        PZE 0                                                        DE6FN0170
DE6FN   SXD DE6FN-1,1          SAVE 1R                               DE6FN0180
        SXD DE6FN-2,2                                                DE6FN0190
        SXD DE6FN-3,4                                                DE6FN0200
        CLA 5,4                                                      DE6FN0210
        STA DUX1                                                     DE6FN0220
        CLA 6,4                                                      DE6FN0230
        STA BCOM               LOC B                                 DE6FN0240
        CLA 7,4                                                      DE6FN0250
        STA RCOM               LOC R                                 DE6FN0260
BCOM    CLA *                                                        DE6FN0270
        STA CALL+1                                                   DE6FN0280
        ARS 18                                                       DE6FN0290
        STO CELM1                                                    DE6FN0300
RCOM    CLA *                                                        DE6FN0310
        ADD CELM1                                                    DE6FN0320
        STO CALL+2             2ND WORD                              DE6FN0330
        CLA 8,4                                                      DE6FN0340
        STA *+1                                                      DE6FN0350
        CLA *                                                        DE6FN0360
        STO CALL+3             3RD WORD                              DE6FN0370
        CLA 9,4                                                      DE6FN0380
        STA *+1                                                      DE6FN0390
        CLA *                                                        DE6FN0400
        STO CALL+4             4TH WORD                              DE6FN0410
        CLA 10,4                                                     DE6FN0420
        STA *+1                                                      DE6FN0430
        CLA *                                                        DE6FN0440
        STO CALL+5             5TH WORD                              DE6FN0450
        CLA 11,4                                                     DE6FN0460
        STA *+1                                                      DE6FN0470
        CLA *                                                        DE6FN0480
        STO CALL+6             6TH WORD                              DE6FN0490
```

```
        CLA  2,4        SET SIGN             DE6FN050
        STA  *+1                             DE6FN051
        CLA  *                               DE6FN052
        TPL  *+3        2ND WORD             DE6FN053
        CLS  CALL+2                          DE6FN054
        STO  CALL+2                          DE6FN055
        CLA  4,4        LOC ENDT             DE6FN056
        STA  NTOTA                           DE6FN057
        CLA  3,4                             DE6FN058
        STA  REVER+3                         DE6FN059
        STA  REVER+5                         DE6FN060
        STA  NTOT                            DE6FN061
        STA  COMN                            DE6FN062
        STA  COMCEL                          DE6FN063
        STA  SVTD                            DE6FN064
NTOTA   CLA  *                               DE6FN065
        ARS  18                              DE6FN066
NTOT    STO  *                               DE6FN067
COMCEL  LDQ  *          FIND TOP T           DE6FN068
        MPY  TRTY                            DE6FN069
        STQ  CELM1                           DE6FN070
        CLA  SVTD                            DE6FN071
        SUB  CELM1                           DE6FN072
        SUB  TWO                             DE6FN073
        ADM  CALL+1                          DE6FN074
        STA  START      SAP T                DE6FN075
        STO  CALL+1                          DE6FN076
        CLA  1,4                             DE6FN077
        STA  *+1                             DE6FN078
        CLA  *                               DE6FN079
        TPL  *+3        SET SIGN PO          DE6FN080
        CLS  CALL+1                          DE6FN081
        STO  CALL+1                          DE6FN082
COMN    LDQ  *          COMPUTE              DE6FN083
        MPY  FIFTN      MID POINT            DE6FN084
        STQ  CELM1                           DE6FN085
        CLA  CELM1                           DE6FN086
        ADD  ONE                             DE6FN087
        STO  HN                              DE6FN088
        CLA  START                           DE6FN089
        ADD  HN                              DE6FN090
        STA  REVER+2                         DE6FN091
        STA  REVER+4                         DE6FN092
        TSX  REVER,4                         DE6FN093
CALL    TSX  DE6F,4     ENTER                DE6FN094
        PZE  ,,DUX                           DE6FN095
        PZE  *                               DE6FN096
        PZE  *                               DE6FN097
        PZE  *                               DE6FN098
        PZE  *                               DE6FN099
        PZE  *                               DE6FN100
```

```
       TSX  REVER,4                                      DE6FN1010
       LXD  DE6FN-1,1                                    DE6FN1020
       LXD  DE6FN-2,2                                    DE6FN1030
       LXD  DE6FN-3,4                                    DE6FN1040
       TRA  12,4                                         DE6FN1050
REVER  LXA  HN,1             REVERSE T BLOCK             DE6FN1060
       LXD  ZERO,2                                       DE6FN1070
IN     LDQ  *,1                                          DE6FN1080
       CLA  *,2                                          DE6FN1090
       STO  *,1                                          DE6FN1100
       STQ  *,2                                          DE6FN1110
       TXI  *+1,2,1                                      DE6FN1120
       TIX  IN,1,1                                       DE6FN1130
       TRA  1,4                                          DE6FN1140
DE6FP1 SXD  DE6FN-1,1                                    DE6FN1150
       SXD  DE6FN-2,2                                    DE6FN1160
       SXD  DE6FN-3,4                                    DE6FN1170
       TSX  REVER,4                                      DE6FN1180
       TSX  DE6F+1,4                                     DE6FN1190
       STO  CELM1                                        DE6FN1200
       TSX  REVER,4                                      DE6FN1210
       LXD  DE6FN-1,1                                    DE6FN1220
       LXD  DE6FN-2,2                                    DE6FN1230
       LXD  DE6FN-3,4                                    DE6FN1240
       CLA  1,4                                          DE6FN1250
       STA  *+2                                          DE6FN1260
       CLA  CELM1                                        DE6FN1270
       STO  *                                            DE6FN1280
       TRA  2,4                                          DE6FN1290
DE6FP2 SXD  DE6FN-1,1                                    DE6FN1300
       SXD  DE6FN-2,2                                    DE6FN1310
       SXD  DE6FN-3,4                                    DE6FN1320
       TSX  REVER,4                                      DE6FN1330
       TSX  DE6F+16,4                                    DE6FN1340
       TSX  REVER,4                                      DE6FN1350
       LXD  DE6FN-1,1                                    DE6FN1360
       LXD  DE6FN-2,2                                    DE6FN1370
       LXD  DE6FN-3,4                                    DE6FN1380
       TRA  1,4                                          DE6FN1390
DE6FP3 SXD  DE6FN-1,1                                    DE6FN1400
       SXD  DE6FN-2,2                                    DE6FN1410
       SXD  DE6FN-3,4                                    DE6FN1420
       TSX  REVER,4                                      DE6FN1430
       LXD  DE6FN-3,4                                    DE6FN1440
       CLA  1,4                                          DE6FN1450
       STA  *+1                                          DE6FN1460
       CLA  *                                            DE6FN1470
       TSX  DE6F+22,4                                    DE6FN1480
       TSX  REVER,4                                      DE6FN1490
       LXD  DE6FN-1,1                                    DE6FN1500
       LXD  DE6FN-2,2                                    DE6FN1510
```

```
        LXD DE6FN-3,4
        TRA 2,4
DE6FP4  SXD DE6FN-1,1
        SXD DE6FN-2,2
        SXD DE6FN-3,4
        TSX REVER,4
        TSX DE6F+588,4
        TSX REVER,4
        LXD DE6FN-1,1
        LXD DE6FN-2,2
        LXD DE6FN-3,4
        TRA 1,4
DE6FNZ  CLA 1,4
        STA *+1
        CLA *
        STO DE6F+500
        TRA 2,4
DE6FPS  CLA DE6F+501
        SSP
        STO DE6F+501
        TRA 1,4
DE6FNG  CLA DE6F+501
        SSM
        STO DE6F+501
        TRA 1,4
DE6FMN  CLA 1,4
        STA *+1
        CLA *
        STO DE6F+509
        TRA 2,4
DE6FMA  CLA 1,4
        STA *+1
        CLA *
        STO DE6F+510
        TRA 2,4
DUX     SXD DUX2,4
        TSX REVER,4
DUX1    TSX 0,4
        TSX REVER,4
        LXD DUX2,4
        TRA 1,4
DUX2    PZE
DE6FCM  CLA 1,4
        STA *+1
        CLA *
        STO DE6F+511
        TRA 2,4
DE6FCH  CLA 1,4
        STA LINKA
        CLA 2,4
        STA LINKB
```

```
LINKA  CLA *                                                                   DE6FN2030
       STO DE6F+516                                                            DE6FN2040
LINKB  CLA *                                                                   DE6FN2050
       STO DE6F+517                                                            DE6FN2060
       TRA 3,4                                                                 DE6FN2070
ZERO   PZE 0                                                                   DE6FN2080
ONE    DEC 1                                                                   DE6FN2090
TWO    DEC 2                                                                   DE6FN2100
 FIFTN DEC 15                                                                  DE6FN2110
 HN    PZE 0                                                                   DE6FN2140
 SVTD  PZE 0                                                                   DE6FN2160
COMMON BSS 44                                                                  DE6FN2170
DE6F   TRA DE6F+0799     COWELL METHOD INTEGRATION SUB (TR SETUP)              DE6FN2180
       SXD DE6F+0504,1             2ND ENTRY, BEGIN                            DE6FN2190
       SXD DE6F+0505,2        INTEGRATION                                      DE6FN2200
       SXD DE6F+0506,4                                                         DE6FN2210
       CLA DE6F+0500     TEST SWITCH FOR COORD. CHANGE                         DE6FN2220
       TZE DE6F+0011   0=NORMAL CASE                                           DE6FN2230
       CLA DE6F+0502      8(-1)0                                               DE6FN2240
       SUB DE6F+0531     1                                                     DE6FN2250
       STO DE6F+0502                                                           DE6FN2260
       TNZ DE6F+0011   CONTINUE IF NOT ZERO                                    DE6FN2270
       TRA DE6F+0091     TR FOR COORD. CHANGE                                  DE6FN2280
       CLA DE6F+0503     TEST R.K. OR COWELL                                   DE6FN2290
       TMI DE6F+0185    -=COWELL                                               DE6FN2300
       CLA DE6F+0709        TEST 1ST TIME FOR R.K.=+                           DE6FN2310
       TMI DE6F+0043    =2N=3RD,...,RTH R.K. INTEGR.                           DE6FN2320
       TRA DE6F+0030   TR TO BEGIN RUNGE KUTTA SERIES (R)                      DE6FN2330
       SXD DE6F+0504,1              PROGRAMMER MAY ENTER HERE FOR              DE6FN2340
       SXD DE6F+0505,2              SPECIAL COORD. CHANGE, 8 SETS              DE6FN2350
       SXD DE6F+0506,4              OF Y,S MUST BE IN T+11N+3                  DE6FN2360
       STZ DE6F+0502      AND T MUST BE AT 8TH Y.                              DE6FN2370
       SXD DE6F+0500,4             8 SETS OF Y PRIME MUST BE                   DE6FN2380
       TRA DE6F+0091     AT T+3N+3 IF F(X,Y,Y PRIME) USED.                     DE6FN2390
       SXD DE6F+0504,1        PROGRAMMER MAY ENTER HERE WITH H IN AC           DE6FN2400
       SXD DE6F+0505,2             AND START NEW RUNGE KUTTA SERIES            DE6FN2410
       SXD DE6F+0506,4             H/R WILL BE INTEGRATION STEP.               DE6FN2420
       STO COMMON+000         NEW H                                            DE6FN2430
       CLA DE6F+0500     CANNOT CHANGE H IF COORD.                             DE6FN2440
       TNZ DE6F+0006   CHANGE IS IN PROGRESS                                   DE6FN2450
       LDQ COMMON+000         NEW H                                            DE6FN2460
       TSX DE6F+0077,2            CHANGE VMIN, VMAX, WMAX FOR H2               DE6FN2470
       CLA DE6F+0519     H/R=R.K. H FOR EACH STEP                              DE6FN2480
       FDP DE6F+0517     R=TOTAL R.K. STEPS                                    DE6FN2490
       STQ 0                 (T+2)                                             DE6FN2500
       LDQ DE6F+0519     DELTA (T) = H                                         DE6FN2510
       FMP DE6F+0519                                                           DE6FN2520
       STO DE6F+0520     DELTA (T) SQUARED                                     DE6FN2530
       STZ DE6F+0518      SET K=0,(K=0(1)6)                                    DE6FN2540
       SSM                                                                     DE6FN2550
       STO DE6F+0709          FIX SWITCH FOR 2ND, ETC ENTRIES                  DE6FN2560
```

```
LXA DE6F+0527,1            7N(FIX TO SAVE 2ND DERIV.)          DE6FN25
SXD DE6F+0527,1            7N(=N)N                             DE6FN25
CLA DE6F+0516       R=R.K. STEPS PER DELTA(T)                  DE6FN25
STO DE6F+0515       R IN DECR.                                 DE6FN26
LXD DE6F+0515,1            R(=1)1                              DE6FN26
CLA DE6F+0516       R=TOTAL R.K. STEPS PER D(T)                DE6FN26
SUB DE6F+0515       R(I)=CURRENT RI=R(=1)1 IN DECR.            DE6FN26
TNZ DE6F+0054      TR IF NOT 1ST R                             DE6FN26
LXA DE6F+0710,4            N                                   DE6FN26
LXD DE6F+0527,2            7N(=N)N                             DE6FN26
CLA 0,4                 T+3N+3(2ND DERIV.)                     DE6FN26
STO 0,2                 T+11N+3+(7N),(7N 2ND DERIVS)           DE6FN26
TXI DE6F+0052,2,=1                                             DE6FN26
TIX DE6F+0049,4,1                                              DE6FN27
SXD DE6F+0527,2            7N(=N)N                             DE6FN27
LXA DE6F+0518,2                                                DE6FN27
TXH DE6F+0122,2,5          IF K=6, TR TO COWELL                DE6FN27
TSX DE6F+0588,4         TR TO RUNGE KUTTA INTEGRATION          DE6FN27
TIX DE6F+0070,1,1          REDUCE R TIL R=RN                   DE6FN27
LXD DE6F+0516,1                R=RN (RESTORE R FOR NEXT CYCLE) DE6FN27
CLA DE6F+0518        K=0(1)6                                   DE6FN27
ADD DE6F+0531        1                                         DE6FN27
STO DE6F+0518                                                  DE6FN27
SUB DE6F+0533        3                                         DE6FN28
TNZ DE6F+0070       SAVE Y(3) AND Y(3) PRIME IF 0              DE6FN28
LXA DE6F+0710,2            N                                   DE6FN28
CLA 0,2                 T+N+3=Y(I)=DEPENDENT VAR. (YI)         DE6FN28
STO 0,2                 T+10N+3=Y(3)                           DE6FN28
CLA 0,2                 T+2N+3=Y(I) PRIME=1ST DERIV.           DE6FN28
STO 0,2                 T+11N+3=Y(3)PRIME                      DE6FN28
TIX DE6F+0065,2,1                                              DE6FN28
SXD DE6F+0515,1            SAVE R                              DE6FN28
SSP                     +=RUNGE KUTTA                          DE6FN28
STO DE6F+0503       ==COWELL METHOD                            DE6FN29
LXD DE6F+0506,4                                                DE6FN29
LXD DE6F+0504,1            1 R.K. STEP COMPLETED               DE6FN29
LXD DE6F+0505,2                                                DE6FN29
TRA 1,4                 RETURN TO PROGRAMMER                   DE6FN29
CLA DE6F+0519        OLD H (SUB. FOR NEW VMIN, VMAX, WMAX)     DE6FN29
STO COMMON+001          H1=OLD H (SAVE FOR CHANGE OF HMAX)     DE6FN29
CLA DE6F+0520        OLD H SQ.=H1 SQ.                          DE6FN29
STO COMMON+000          H1 SQ.                                 DE6FN29
STQ DE6F+0519        NEW H=H2                                  DE6FN29
FMP DE6F+0519                                                  DE6FN30
STO DE6F+0520        NEW H SQ.=H2 SQ.                          DE6FN30
LXA DE6F+0533,5         3 IN IR1, IR4                          DE6FN30
CLA COMMON+000          H1 SQ.                                 DE6FN30
FDP DE6F+0520        H2 SQ.                                    DE6FN30
FMP DE6F+0515,4         CHANGE VMIN, VMAX, VMAX                DE6FN30
STO DE6F+0515,4         (V/H1 SQ.)X(H1 SQ)/(H2 SQ.)            DE6FN30
TIX DE6F+0085,4,1                                              DE6FN30
```

```
TRA 1,2                                                                DE6FN3080
LDQ 0                   T+2=H (COORD. CHANGE BEGINS)                   DE6FN3090
FMP DE6F+0544        =8.                                               DE6FN3100
FAD 0                  T+1=T                                           DE6FN3110
STO 0                  T+1                                             DE6FN3120
LXA DE6F+0528,1                  8N                                    DE6FN3130
SXD DE6F+0528,1                  8N(-N)0                               DE6FN3140
CLA 0                  T+1=(T)                                         DE6FN3150
FAD 0                  T+2=H                                           DE6FN3160
STO 0                  T+1=(T+H)                                       DE6FN3170
LXA DE6F+0710,2             N                                          DE6FN3180
CLA 0,1                 T+11N+3+(8N)=(Y0,Y1,....,Y7)                   DE6FN3190
STO 0,2                T+N+3                                           DE6FN3200
CLA 0,1                 T+3N+3+(8N)=1ST DERIV.                         DE6FN3210
STO 0,2                 T+2N+3(IF SAVED)                               DE6FN3220
TXI DE6F+0106,1,-1          MOVE DEPENDENT VAR. TO T+3 THRU T+N+2      DE6FN3230
TIX DE6F+0101,2,1                                                      DE6FN3240
TSX 0,4                V (FORM2ND DERIV.)                              DE6FN3250
LXD DE6F+0528,1                  8N                                    DE6FN3260
LXA DE6F+0710,2             N                                          DE6FN3270
CLA 0,2                T+3N+3 (2ND DERIV.)                             DE6FN3280
STO 0,1                 T+3N+3+(8N)                                    DE6FN3290
TXI DE6F+0113,1,-1              STORE 2ND DERIV. (8 SETS)              DE6FN3300
TIX DE6F+0110,2,1                                                      DE6FN3310
TXH DE6F+0096,1,0               LOOP FOR 8 SETS OF Y                   DE6FN3320
LDQ 0                   (T+2)=H=DELTA(T)                               DE6FN3330
TSX DE6F+0077,2                 CHANGE VMIN, VMAX, WMAX                DE6FN3340
CLA DE6F+0519        H2=NEW H                                          DE6FN3350
FDP COMMON+001          H1=OLD H                                       DE6FN3360
FMP DE6F+0512,1         CHANGE UNITS OF HMIN,HMAX,                     DE6FN3370
STO DE6F+0512,1         AND XMIN                                       DE6FN3380
TIX DE6F+0117,1,1                                                      DE6FN3390
STZ DE6F+0521       CLEAR W, W=MAX (DELTA VII)/Y                       DE6FN3400
STZ DE6F+0522       CLEAR V, V=MAX (DELTA VI)/Y                        DE6FN3410
LXA DE6F+0529,4                 11N                                    DE6FN3420
SXD DE6F+0529,4                                                        DE6FN3430
LXA DE6F+0710,4             N                                          DE6FN3440
LXA DE6F+0527,2                  7N                                    DE6FN3450
SXD DE6F+0527,2                  7N(-1)6N                              DE6FN3460
CLA 0,4                 T+N+3=Y                                        DE6FN3470
STO DE6F+0711          SEE V AND W CALC.                               DE6FN3480
CLA DE6F+0502        NORMALLY=8 (NO CHANGE IN COORD.)                  DE6FN3490
TNZ DE6F+0316       0=COORD. CHANGE                                    DE6FN3500
TSX DE6F+0380,2                  FORM DIFF.                            DE6FN3510
PZE 0                   T+3N+3+(7N)                                    DE6FN3520
CLA DE6F+0520        H SQ.                                             DE6FN3530
TSX DE6F+0344,2                 FORM SUM OF PRODUCTS                   DE6FN3540
PZE DE6F+0552,0,COMMON+21       C(I)XDELTAS                            DE6FN3550
CLA 0,4                 T+15N+3(Y3,COORD. CHANGE)                      DE6FN3560
TSX DE6F+0482,2                  ADD Y3 TO C+3,C+4                     DE6FN3570
TSX DE6F+0373,2                  F4(II) IN A1 AND A2                   DE6FN3580
```

```
        CLS DE6F+0507                                            DE6FN359
        STO DE6F+0714         -F4(II)                            DE6FN360
        CLS DE6F+0508                                            DE6FN361
        STO DE6F+0715                                            DE6FN362
        TSX DE6F+0380,2              FORM DIFF.                  DE6FN363
        PZE 0                 T+4N+3+(7N)                        DE6FN364
        TSX DE6F+0345,2         H SQ. STILL IN COM+2             DE6FN365
        PZE DE6F+0552,0,COMMON+21                                DE6FN366
        CLA 0,4               T+16N+3,(Y4, COORD. CHANGE)        DE6FN367
        TSX DE6F+0482,2              ADD Y4 TO C+3,C+4           DE6FN368
        TSX DE6F+0373,2              F5(II) IN A1 AND A2         DE6FN369
        CLA DE6F+0714                                            DE6FN370
        STO COMMON+003        -F4(II)                            DE6FN371
        CLA DE6F+0715                                            DE6FN372
        STO COMMON+004                                           DE6FN373
        TSX DE6F+0491,2            F4(I)=F5(II)-F4(II)           DE6FN374
        STO COMMON+003        F4(I)                              DE6FN375
        STQ COMMON+004                                           DE6FN376
        CLA COMMON+008        Y4 (2ND DERIV.)                    DE6FN377
        TSX DE6F+0482,2              F5(I) IN COMMON+3,4         DE6FN378
        LXA DE6F+0533,1              3                           DE6FN379
        TSX DE6F+0491,2            F5(II) OR F6(II) IF COORD. CHG.  DE6FN380
        STO DE6F+0507                                            DE6FN381
        STQ DE6F+0508                                            DE6FN382
        CLA COMMON+012,1      Y(4),Y(5),Y(6) 2ND DERIV.          DE6FN383
        TSX DE6F+0482,2              OR Y(5),Y(6),Y(7) IF COORD. CHANGE DE6FN384
        TSX DE6F+0491,2            FORM F5(II),F6(II),F7(II),F8(II)  DE6FN385
        TIX DE6F+0163,1,1          OR F6(II),F7(II),F8(II),F9(II)  DE6FN386
                                                                 DE6FN387
        STO COMMON+027        F8(II) OR F9(II)                   DE6FN388
        STQ COMMON+028                                           DE6FN389
        CLA COMMON+003        F7(I) OR F8(I)                     DE6FN390
        STO COMMON+029                                           DE6FN391
        CLA COMMON+004                                           DE6FN392
        STO COMMON+030                                           DE6FN393
        LXD DE6F+0529,1              11N(-11)11                  DE6FN394
        LXA DE6F+0541,2              11                          DE6FN395
        CLA COMMON+031,2      SAVE DIFF. AND F8(II),F7(I)        DE6FN396
        STO 0,1               T+19N+3+(11N)                      DE6FN397
        TXI DE6F+0180,1,-1                                       DE6FN398
        TIX DE6F+0177,2,1                                        DE6FN399
        SXD DE6F+0529,1              SAVE FOR NEXT SET           DE6FN400
        LXD DE6F+0527,2                                          DE6FN401
        TXI DE6F+0184,2,-1                                       DE6FN402
        TIX DE6F+0128,4,1          COMPLETE N EQNS.              DE6FN403
        TSX DE6F+0414,4              TEST H (MAY TR R.K. AGAIN)  DE6FN404
        CLA DE6F+0502         8(-1)0   IF COORD CHG. (NORMALLY=8)  DE6FN405
        TNZ DE6F+0191       IF NOT=0, STILL IN COORD. CHG. OR NORMAL  DE6FN406
        CLA DE6F+0538         8 RESTORE COUNTER FOR              DE6FN407
        STO DE6F+0502         NEXT CHANGE IN COORD.              DE6FN408
        STZ DE6F+0500         RESTORE SWCH TO NORMAL             DE6FN409
```

DE6FN (Cont'd)

```
CLA DE6F+0519       H                                          DE6FN4100
STO                       T+2=(H) RESTORE AFTER R.K.           DE6FN4110
FAD                       T+1=T=INDEPENDENT VAR.               DE6FN4120
STO COMMON+000            T+H/2 (MOST SIG.)                    DE6FN4130
STQ COMMON+001            LST. SIG.                            DE6FN4140
CLA COMMON+001                                                 DE6FN4150
FAD DE6F+0718             CUM. LST. SIG.                       DE6FN4160
FAD COMMON+000                                                 DE6FN4170
STQ DE6F+0718             SAVE LST. SIG. FOR NEXT INTEGRATION  DE6FN4180
STO                       T+1=(T+H)                            DE6FN4190
LXA DE6F+0710,4               N                                DE6FN4200
LXA DE6F+0529,1                  11N                           DE6FN4210
CLA DE6F+0530       T+19N+3+(7)                                DE6FN4220
STA DE6F+0208                                                  DE6FN4230
STA DE6F+0218      T+19N+3+(7)                                 DE6FN4240
CLA DE6F+0520                                                  DE6FN4250
TSX DE6F+0344,2                                                DE6FN4260
PZE 0,0,DE6F+0566         N+3+(7) STORED ABOVE (+11N)          DE6FN4270
LDQ DE6F+0520       H SQ                                       DE6FN4280
FMP 0,1                   T+19N+3+(11N+7)=F8(II)               DE6FN4290
TSX DE6F+0473,2                                                DE6FN4300
FAD COMMON+003                                                 DE6FN4310
STO 0,4                   T+N+3=NEW Y                          DE6FN4320
CLA DE6F+0717               TEST MISSING 1ST DERIV. IN F       DE6FN4330
TMI DE6F+0224      -=F(X,Y),+=F(X,Y,Y PRIME)                   DE6FN4340
CLA DE6F+0519       H                                          DE6FN4350
TSX DE6F+0344,2                                                DE6FN4360
PZE 0,0,DE6F+0587         N+3+(7) (+11N)   INCR. BY 11         DE6FN4370
LDQ DE6F+0519       H                                          DE6FN4380
FMP 0,1                   T+19N+3+(11N+9)=F7(I)                DE6FN4390
TSX DE6F+0473,2                                                DE6FN4400
FAD COMMON+003                                                 DE6FN4410
STO 0,4                   T+2N+3=NEW Y PRIME                   DE6FN4420
CAL DE6F+0208      T+19N+3+(7)                                 DE6FN4430
ADM DE6F+0541       11                                         DE6FN4440
STA DE6F+0208                                                  DE6FN4450
STA DE6F+0218                                                  DE6FN4460
TXI DE6F+0229,1,-11              SET UP NEXT F8(II),F7(I)      DE6FN4470
TIX DE6F+0206,4,1             LOOP FOR N EQNS.                 DE6FN4480
TSX 0,4                   V (TR TO FORM 2ND DERIV.)            DE6FN4490
LXA DE6F+0529,1                  11N                           DE6FN4500
LXA DE6F+0710,4               N                                DE6FN4510
LXA DE6F+0541,2                  11                            DE6FN4520
CLA 0,1                   T+19N+3+(11N)                        DE6FN4530
STO COMMON+031,2          READ DIFF. TO COM+20                 DE6FN4540
TXI DE6F+0237,1,-1                                             DE6FN4550
TIX DE6F+0234,2,1                                              DE6FN4560
SXD DE6F+0529,1                                                DE6FN4570
CLA 0,4                   T+3N+3=Y(7) 2ND DERIV.               DE6FN4580
STO COMMON+039            Y(7) 2ND DERIV.                      DE6FN4590
LXA DE6F+0536,1                  6                             DE6FN4600
```

DE6FN (Cont'd)

```
LXA DE6F+0571,2        ZERO TO IR2                                     DE6FN461
FSB COMMON+026,2       Y(6) 2ND DERIV. D5(I)...D0(VI)                  DE6FN462
STO COMMON+038,2       D6(I),D5(II),D4(III),...D1(VI)                  DE6FN463
TXI DE6F+0246,2,1          WE ARE GOING BACKWARDS HERE                 DE6FN464
TIX DE6F+0243,1,1                                                      DE6FN465
CLA DE6F+0520      H SQ.                                               DE6FN466
TSX DE6F+0344,2            B(I) X DELTAS                               DE6FN467
PZE DE6F+0573,0,COMMON+40                                              DE6FN468
LDQ DE6F+0520      H SQ.                                               DE6FN469
FMP COMMON+027         F8(II)                                          DE6FN470
TSX DE6F+0473,2                                                        DE6FN471
FAD COMMON+003                                                         DE6FN472
STO 0,4                T+N+3=NEW Y                                     DE6FN473
CLA DE6F+0500      TEST COORD. CHANGE                                  DE6FN474
TNZ DE6F+0258     CANNOT SAVE IF COORD. CHANGE                         DE6FN475
STQ 0,4                T+9N+3=LST. SIG. (FOR R.K.)                     DE6FN476
CLA DE6F+0519      H                                                   DE6FN477
TSX DE6F+0344,2            B(I) DOT Y DELTAS                           DE6FN478
PZE DE6F+0580,0,COMMON+40                                              DE6FN479
LDQ DE6F+0519      H                                                   DE6FN480
FMP COMMON+029         F7(I)                                           DE6FN481
TSX DE6F+0473,2                                                        DE6FN482
FAD COMMON+003                                                         DE6FN483
STO 0,4                T+2N+3=NEW Y PRIME                              DE6FN484
CLA DE6F+0500                                                          DE6FN485
TNZ DE6F+0269                                                          DE6FN486
STQ 0,4                T+5N+3=LST. SIG. (FOR R.K.)                     DE6FN487
LXD DE6F+0529,1                                                        DE6FN488
TIX DE6F+0233,4,1          COMPLETE N SETS                             DE6FN489
TSX 0,4                V (FORM Y(7) 2ND DERIV.                         DE6FN490
STZ DE6F+0522      CLEAR V                                             DE6FN491
STZ DE6F+0521      CLEAR W                                             DE6FN492
LXA DE6F+0529,4            11N                                         DE6FN493
LXA DE6F+0710,1          N                                             DE6FN494
SXD DE6F+0529,4            11N(-11)11                                  DE6FN495
LXA DE6F+0541,2            11                                          DE6FN496
CLA 0,4                T+19N+3+(11N)                                   DE6FN497
STO COMMON+031,2       READ DIFF. TO COM+20                            DE6FN498
TXI DE6F+0281,4,-1                                                     DE6FN499
TIX DE6F+0278,2,1                                                      DE6FN500
CLA 0,1                T+3N+3=Y(7) 2ND DERIV.                          DE6FN501
STO COMMON+039         Y(7) 2ND DERIV.                                 DE6FN502
CLA COMMON+029         F7(I) MOST SIG.                                 DE6FN503
STO COMMON+003                                                         DE6FN504
CLA COMMON+030         F7(I) LST. SIG.                                 DE6FN505
STO COMMON+004                                                         DE6FN506
CLA COMMON+039         Y(7) 2ND DERIV.                                 DE6FN507
TSX DE6F+0482,2                                                        DE6FN508
STO COMMON+042         F8(I) MOST SIG.                                 DE6FN509
STQ COMMON+043         F8(I) LST. SIG.                                 DE6FN510
CLA COMMON+027         F8(II) MOST. SIG.                               DE6FN511
```

DE6FN (Cont'd)

```
STO DE6F+0507                                                     DE6FN5120
CLA COMMON+028          F8(II) LST. SIG.                          DE6FN5130
STO DE6F+0508                                                     DE6FN5140
TSX DE6F+0491,2              DOUBLE PREC. FAD                     DE6FN5150
STO COMMON+040          F9(II) MOST. SIG.                         DE6FN5160
STQ COMMON+041          F9(II) LST. SIG.                          DE6FN5170
LXA DE6F+0536,4              6                                    DE6FN5180
LXA DE6F+0571,2                                                   DE6FN5190
CLA COMMON+039,2        Y(7) 2ND DERIV.,D6(I),....,D1(VI)         DE6FN5200
FSB COMMON+026,2        Y(6) 2ND DERIV.,D5(I),....,D0(VI)         DE6FN5210
STO COMMON+038,2        D6(I),D5(II),...D1(VI)                    DE6FN5220
TXI DE6F+0305,2,1            WE ARE GOING BACKWARDS HERE          DE6FN5230
TIX DE6F+0301,4,1                                                 DE6FN5240
LXD DE6F+0529,4                11N(-11)11                         DE6FN5250
LXA DE6F+0541,2                11                                 DE6FN5260
CLA COMMON+044,2        SAVE FOR NEXT INTEGRATION                 DE6FN5270
STO 0,4                 T+19N+3+(11N)                             DE6FN5280
TXI DE6F+0311,4,-1                                                DE6FN5290
TIX DE6F+0308,2,1                                                 DE6FN5300
TXH DE6F+0340,1,*             N-B IN DECR.                        DE6FN5310
TIX DE6F+0276,1,1             COMPLETE N SETS                     DE6FN5320
SSM                     --COWELL STEP                             DE6FN5330
TRA DE6F+0072      RETURN TO PROGRAMMER                           DE6FN5340
TSX DE6F+0380,2                 NORMAL CASE (NO COORD. CHANGE)    DE6FN5350
PZE 0                   T+11N+3+(7N) (FORM DIFF.)                 DE6FN5360
CLA DE6F+0519      H                                              DE6FN5370
TSX DE6F+0344,2               FORM SUM OF PRODUCTS                DE6FN5380
PZE DE6F+0559,0,COMMON+21       D(I)XDELTAS                       DE6FN5390
CLA 0,4                 T+11N+3 (Y3 DOT)                          DE6FN5400
TSX DE6F+0482,2                 ADD Y3 DOT TO C+3,C+4             DE6FN5410
TSX DE6F+0373,2                 F4(I) IN A1 AND A2                DE6FN5420
CLA DE6F+0507                                                     DE6FN5430
STO DE6F+0714          F4(I)                                      DE6FN5440
CLA DE6F+0508                                                     DE6FN5450
STO DE6F+0715                                                     DE6FN5460
CLA DE6F+0520      H SQ.                                          DE6FN5470
TSX DE6F+0344,2               FORM SUM OF PRODUCTS                DE6FN5480
PZE DE6F+0552,0,COMMON+21       D(I) X DELTAS                     DE6FN5490
CLA 0,4                 T+10N+3 (Y3)                              DE6FN5500
TSX DE6F+0482,2                 ADD Y3 TO C+3,C+4                 DE6FN5510
TSX DE6F+0373,2                 F4(II) IN A1 AND A2               DE6FN5520
CLA DE6F+0714          F4(I)                                      DE6FN5530
STO COMMON+003                                                    DE6FN5540
CLA DE6F+0715                                                     DE6FN5550
STO COMMON+004                                                    DE6FN5560
TXH DE6F+0453,4,*               N-B IN DECR. (TEST 1ST B EQNS)    DE6FN5570
TRA DE6F+0161                                                     DE6FN5580
CLA 0,1                T+N+3=Y                                    DE6FN5590
STO DE6F+0711          Y                                          DE6FN5600
TSX DE6F+0455,2              FORM MAX V, W                        DE6FN5610
TRA DE6F+0313      FOR THIS Y                                     DE6FN5620
```

DE6FN (Cont'd)

```
       STO COMMON+002            H SQ. OR H IN AC                                DE6FN563
       SXD DE6F+0707,1                SUB. TO FORM (CONSTANTS X DIFF.)           DE6FN564
       SXD DE6F+0708,2                                                           DE6FN565
       CAL 1,2                   A,0,B (A=CONSTANTS, B=DIFF.)                    DE6FN566
       STA DE6F+0356          CONSTANT ADDR. +7                                  DE6FN567
       ARS 18                                                                    DE6FN568
       STA DE6F+0357          DIFF. ADDR. +7                                     DE6FN569
       CLA COMMON+008            Y3 2ND DERIV.                                   DE6FN570
       STO COMMON+014            (OR Y4 IF COORD. CHANGE)                        DE6FN571
       STZ COMMON+003                                                            DE6FN572
       STZ COMMON+004                                                            DE6FN573
       LXA DE6F+0537,1                7                                          DE6FN574
       LDQ 0,1                   CO+7, OR DO+7, OR N6+7, ETC.                    DE6FN575
       FMP 0,1                   COMMON+21, OR COM+27, ETC.                      DE6FN576
       TSX DE6F+0473,2                ACCUM. PRODUCT IN C+3,C+4                  DE6FN577
       TIX DE6F+0356,1,1                                                         DE6FN578
       LDQ COMMON+002            H SQ. OR H                                      DE6FN579
       FMP COMMON+003            MOST SIG.                                       DE6FN580
       STO COMMON+003                                                            DE6FN581
       STQ COMMON+000                                                            DE6FN582
       LDQ COMMON+002            H SQ. OR H                                      DE6FN583
       FMP COMMON+004                                                            DE6FN584
       FAD COMMON+000                                                            DE6FN585
       FAD COMMON+003                                                            DE6FN586
       STO COMMON+003            MOST SIG. IN COMMON+3                           DE6FN587
       STQ COMMON+004            LST. SIG. IN COMMON+4                           DE6FN588
       LXD DE6F+0707,1                                                           DE6FN589
       LXD DE6F+0708,2                                                           DE6FN590
       TRA 2,2                                                                   DE6FN591
       CLA COMMON+003            DIVIDES C+3,C+4 BY C+2                          DE6FN592
       FDP COMMON+002            H SQ OR H                                       DE6FN593
       STQ DE6F+0507          MOST SIG. IN A1                                    DE6FN594
       FAD COMMON+004            REMAINDER + LST. SIG.                           DE6FN595
       FDP COMMON+002            H SQ. OR H                                      DE6FN596
       STQ DE6F+0508          LST. SIG. IN A2                                    DE6FN597
       TRA 1,2                                                                   DE6FN598
       SXD DE6F+0716,2               MOVE 2ND DERIV. AND FORM DIFF.              DE6FN599
       CAL 1,2                   STARTING ADDR. OF 2ND DERIV.                    DE6FN600
       STA DE6F+0385                                                             DE6FN601
       LXD DE6F+0527,2                7N(-1)6N                                   DE6FN602
       LXA DE6F+0571,1                                                           DE6FN603
       CLA 0,2                   START ADDR. (T+11N+3+(7N), ETC.)                DE6FN604
       STO COMMON+005,1          MOVE 2ND DERIV. TO COM+5                        DE6FN605
       STO COMMON+020,1          AND COM+20                                      DE6FN606
       TXI DE6F+0389,1,-1                                                        DE6FN607
       TIX DE6F+0385,2,*              N IN DECR.                                 DE6FN608
       LXA DE6F+0536,2                6 (FORM DIFFERENCES)                       DE6FN609
       SXD COMMON+000,2                                                          DE6FN610
       LXA DE6F+0571,1           ZERO (2ND DERIV. STORED AT                      DE6FN611
       CLA COMMON+021,1         AT COMMON+5, AGAIN AT COM +20                    DE6FN612
       FSB COMMON+020,1         FORM DIFFERENCES IN                              DE6FN613
```

DE6FN (Cont'd)

```
      STO COMMON+020,1         COM+20 THRU COM+25                DE6FN6140
      TXI DE6F+0397,1,-1                                         DE6FN6150
      TIX DE6F+0393,2,1            LOOP 6,5,....,1 TIMES         DE6FN6160
      LXD COMMON+000,2                                           DE6FN6170
      TXH DE6F+0405,2,5            TR IF 1ST DIFF.               DE6FN6180
      TXH DE6F+0408,2,3            TR IF 2ND OR 3RD DIFF.        DE6FN6190
      TXH DE6F+0411,2,1            TR IF 4TH OR 5TH DIFF.        DE6FN6200
      TIX DE6F+0391,2,1                                          DE6FN6210
      LXD DE6F+0716,2                                            DE6FN6220
      TRA 2,2                                                    DE6FN6230
      CLA COMMON+023           DELTA (I)3                        DE6FN6240
      STO COMMON+015           IN COM+15                         DE6FN6250
      TRA DE6F+0402                                              DE6FN6260
      CLA COMMON+022           DELTA (II)2 OR DELTA (III)2       DE6FN6270
      STO COMMON+021,2         IN COM+16 OR COM+17               DE6FN6280
      TRA DE6F+0402                                              DE6FN6290
      CLA COMMON+021           DELTA(IV)1 OR DELTA(V)1           DE6FN6300
      STO COMMON+021,2         IN COM+18 OR COM+19               DE6FN6310
      TRA DE6F+0402                                              DE6FN6320
      CLA DE6F+0500        SWITCH (0=NORMAL CASE)                DE6FN6330
      TNZ 1,4                 STOP TEST IF NON/ZERO              DE6FN6340
      CLA DE6F+0501         TEST FIXED H                         DE6FN6350
      TMI 1,4                   ==FIXED H INTERGRATION           DE6FN6360
      CLA DE6F+0519        H=DELTA(T) COWELL TEST                DE6FN6370
      FDP DE6F+0719         2.                                   DE6FN6380
      STQ COMMON+000           H/2                               DE6FN6390
      CLA DE6F+0523        002.0 (WILL MULT BY 2 SQUARED         DE6FN6400
      STO COMMON+001           WILL MULT VMAX,VMIN, AND WMAX     DE6FN6410
      CLA DE6F+0509        MINIMUM H ALLOWED (CAN=0)             DE6FN6420
      LRS 0                    SET SIGNS PLUS                    DE6FN6430
      TLQ DE6F+0438        H/2 LESS THAN H MIN.                  DE6FN6440
      CLA DE6F+0513        10 TO (3-S)/(H SQUARED) =MAX V        DE6FN6450
      LDQ DE6F+0522        V=MAX (DELTA VI)/Y                    DE6FN6460
      TLQ DE6F+0438        V LESS THAN MAX. V                    DE6FN6470
      CLA COMMON+000           H/2 OR 2H                         DE6FN6480
      STO DE6F+0519        SET H=H/2 (OR 2H) AND                 DE6FN6490
      LXA DE6F+0533,4                3 (SCALE VMIN, VMAX, AND WMAX)  DE6FN6500
      CLA DE6F+0515,4            VMIN, VMAX, AND WMAX            DE6FN6510
      TZE DE6F+0436        SKIP IF ZERO                          DE6FN6520
      ADD COMMON+001           + OR - 002.0                      DE6FN6530
      STO DE6F+0515,4            NEW VALUES DUE TO CHANGE IN H   DE6FN6540
      TIX DE6F+0432,4,1                                          DE6FN6550
      TRA DE6F+0030      RETURN TO RUNGE KUTTA                   DE6FN6560
      CLA DE6F+0519        H=DELTA(T)                            DE6FN6570
      FAD DE6F+0519                                              DE6FN6580
      STO COMMON+000           2H                                DE6FN6590
      SSP                                                        DE6FN6600
      LDQ DE6F+0510        MAXIMUM H ALLOWED                     DE6FN6610
      TLQ DE6F+0452        2H GREATER THAN H MAX.                DE6FN6620
      LDQ DE6F+0512       .10 TO (-1-S)/(H SQUARED)= VMIN        DE6FN6630
      CLA DE6F+0522        V=MAX(DELTA VII)/Y                    DE6FN6640
```

DE6FN- (Cont'd)

```
TLQ DE6F+0452       V GR. THAN V MIN                                    DE6FN665
CLS DE6F+0523       002.0 DIVIDES BY 2 SQUARED                          DE6FN666
STO COMMON+001          DECREASES VMIN, WMAX, VMAX                      DE6FN667
CLA DE6F+0514      .10 TO (-1-S)/(H SQUARED)= HMAX                      DE6FN668
LDQ DE6F+0521       W=MAX (DELTA VII)/Y                                 DE6FN669
TLQ DE6F+0429       W LESS THAN W MAX. (SET H=2H)                       DE6FN670
TRA 1,4                 RETURN TO PROGRAM                               DE6FN671
TSX DE6F+0455,2             FORM LARGEST V AND W                        DE6FN672
TRA DE6F+0339                                                           DE6FN673
CLA DE6F+0501        SUB TO FORM MAX V AND W FOR 1ST N-B EQNS.          DE6FN674
TMI 1,2                   --FIXED H INTEGRATION                         DE6FN675
CLA DE6F+0711            Y                                              DE6FN676
SSP                     SET Y PLUS                                      DE6FN677
LDQ DE6F+0511       A=MIN. Y ALLOWED                                    DE6FN678
TLQ DE6F+0462        TR IF Y GREATER THAN Y MIN.                        DE6FN679
STQ DE6F+0711          OTHERWISE, SET Y=Y MIN.=A                        DE6FN680
SXD COMMON+002,1                                                        DE6FN681
LXA DE6F+0532,1                 2                                       DE6FN682
CLA COMMON+022,1          DELTA VI (OR DELTA V)                         DE6FN683
FDP DE6F+0711          Y (OR Y MIN)                                     DE6FN684
CLA DE6F+0523,1                 MAX. W(OR V) THIS FAR                   DE6FN685
LRS 0                   SET SIGNS PLUS                                  DE6FN686
TLQ DE6F+0470       NEW W (OR V) IS SMALLER                             DE6FN687
STQ DE6F+0523,1                 NEW MAX. W (OR V)                       DE6FN688
TIX DE6F+0464,1,1           LOOP FOR V                                  DE6FN689
LXD COMMON+002,1                                                        DE6FN690
TRA 1,2                                                                 DE6FN691
STQ COMMON+000          LST SIG. (PREC. AND 1/2)                        DE6FN692
FAD COMMON+003          MOST SIG.                                       DE6FN693
STO COMMON+003                                                          DE6FN694
STQ COMMON+001                                                          DE6FN695
CLA COMMON+000                                                          DE6FN696
FAD COMMON+001                                                          DE6FN697
FAD COMMON+004                                                          DE6FN698
STO COMMON+004          LST. SIG.                                       DE6FN699
TRA 1,2                                                                 DE6FN700
UFA COMMON+003           ADDS AC TO COM+3, COM+4                        DE6FN701
STO COMMON+003           AND STORES ANSWER IN C+3,C+4                   DE6FN702
STQ COMMON+000                                                          DE6FN703
CLA COMMON+000                                                          DE6FN704
UFA COMMON+004                                                          DE6FN705
FAD COMMON+003                                                          DE6FN706
STO COMMON+003           MOST SIG. IN COMMON+3                          DE6FN707
STQ COMMON+004           LST. SIG. IN COMMON+4                          DE6FN708
TRA 1,2                                                                 DE6FN709
CLA DE6F+0507       DOUBLE PRECISION FAD                                DE6FN710
UFA COMMON+003                                                          DE6FN711
STO COMMON+001                                                          DE6FN712
STQ COMMON+000                                                          DE6FN713
CLA COMMON+000                                                          DE6FN714
UFA DE6F+0508                                                           DE6FN715
```

DE6FN (Cont'd)

```
 UFA COMMON+004                                                   DE6FN7160
 FAD COMMON+001                                                   DE6FN7170
 TRA 1,2                                                          DE6FN7180
 BSS 1             SWITCH (PROGRAMMER CHANGES) (0=NORMAL)         DE6FN7190
 BSS 1             FIXED H STORAGE (-=FIXED STEP SIZE)            DE6FN7200
 BSS 1             COUNTER FOR CHANGE OF COORD.                   DE6FN7210
 BSS 1             +=R.K. STEP,-=COWELL STEP                      DE6FN7220
 BSS 1             SAVE INDEX REG. FOR (IR1)                      DE6FN7230
 BSS 1             COWELL METHOD       (IR2)                      DE6FN7240
 BSS 1                                 (IR4)                      DE6FN7250
 BSS 1             1ST NO.(MOST SIG.)                             DE6FN7260
 BSS 1             1ST NO. (LST. SIG.)                            DE6FN7270
 BSS 1             MIN. H (ABS. VALUE) PROG. SPEC.                DE6FN7280
 BSS 1              HMAX (IF H=0, SET HMAX= 1E18                  DE6FN7290
 BSS 1             A=MIN. Y ALLOWED FOR TEST                      DE6FN7300
 BSS 1             10 TO (-1-S)/(H SQUARED)=VMIN                  DE6FN7310
 BSS 1             10 TO (3-S)/(H SQUARED)=VMAX                   DE6FN7320
 BSS 1             10 TO (-1-S)/(H SQUARED)=WMAX                  DE6FN7330
 BSS 1             R(I)=CURRENT RI=R(-1)1 IN DECR.                DE6FN7340
 BSS 1             R=TOTAL ITERATIONS PER D(T)                    DE6FN7350
 BSS 1             R (FLOATING PT.) (ABOVE=FIXED PT.)             DE6FN7360
 BSS 1             0(1)8 (COUNTER FOR R.K. LOOPS)                 DE6FN7370
 BSS 1             H=DELTA(T)=CURRENT D(T)                        DE6FN7380
 BSS 1             DELTA (T) SQUARED                              DE6FN7390
 BSS 1             W=MAX (DELTA VII)/Y                            DE6FN7400
 BSS 1             V=MAX(DELTA VI)/Y                              DE6FN7410
 OCT 002000000000  MULT. BY 2 SQUARED (OR DIVIDE)                 DE6FN7420
 OCT 233000000000  NO TO FLOAT FIXED PT NO.                       DE6FN7430
 DEC 1E18           IF H=0, SET HMAX = 1E18                       DE6FN7440
 DEC 0              2N                                            DE6FN7450
 DEC 0              7N                                            DE6FN7460
 DEC 0              8N                                            DE6FN7470
 DEC 0              11N                                           DE6FN7480
 DEC 0              DUMMY=T+19N+3+(7)                             DE6FN7490
 DEC 1             1                                              DE6FN7500
 DEC 2             2                                              DE6FN7510
 DEC 3             3                                              DE6FN7520
 DEC 4             4                                              DE6FN7530
 DEC 5             5                                              DE6FN7540
 DEC 6             6                                              DE6FN7550
 DEC 7              7                                             DE6FN7560
 DEC 8             8                                              DE6FN7570
 DEC 9             9                                              DE6FN7580
 DEC 10            10                                             DE6FN7590
 DEC 11             11                                            DE6FN7600
 DEC 13            13                                             DE6FN7610
 DEC 1.                                                           DE6FN7620
 DEC -8.            -8.                                           DE6FN7630
 DEC -8.333333333E-2  (SIGNS HAVE BEEN REVERSED)  (C0)            DE6FN7640
 DEC 0              C1=0                                          DE6FN7650
 DEC +4.166666667E-3  C2                                          DE6FN7660
```

DE6FN (Cont'd)

```
DEC 0                       C3=0                                         DE6FN76
DEC -5,125661376E-4         C4                                           DE6FN76
DEC 0                       C5=0                                         DE6FN76
DEC +7,964065256E-5         C6                                           DE6FN77
DEC +0,5                   (SIGNS HAVE BEEN REVERSED)  (D0)              DE6FN77
DEC +8,333333333E-2         D1                                           DE6FN77
DEC -4,166666667E-2         D2                                           DE6FN77
DEC -1,527777778E-2         D3                                           DE6FN77
DEC +7,638888889E-3         D4                                           DE6FN77
DEC +3,1580688E-3           D5                                           DE6FN77
DEC -1,5790344E-3           D6                                           DE6FN77
DEC +6,549575617E-2        (NO CHANGE IN SIGNS)    (N6)                  DE6FN77
DEC +6,820436508E-2         N5                                           DE6FN77
DEC +7,134589948E-2         N4                                           DE6FN78
DEC +0,075                  N3                                           DE6FN78
DEC +7,916666667E-2         N2                                           DE6FN78
DEC +8,333333333E-2         N1                                           DE6FN78
DEC +8,333333333E-2         N0                                           DE6FN78
DEC -2,708608907E-3         B6                                           DE6FN78
DEC -3,141534392E-3         B5                                           DE6FN78
DEC -3,654100529E-3         B4                                           DE6FN78
DEC -4,166666667E-3         B3                                           DE6FN78
DEC -4,166666667E-3         B2                                           DE6FN78
DEC 0                       B1=0                                         DE6FN79
DEC +8,333333333E-2         B0                                           DE6FN79
DEC -1,136739418E-2         B6 DOT                                       DE6FN79
DEC -1,426917989E-2         B5 DOT                                       DE6FN79
DEC -1,875E-2               B4 DOT                                       DE6FN79
DEC -2,638888889E-2         B3 DOT                                       DE6FN79
DEC -4,166666667E-2         B2 DOT                                       DE6FN79
DEC -8,333333333E-2         B1 DOT                                       DE6FN79
DEC +0,5                    B0 DOT                                       DE6FN79
DEC ,304224537              (NO CHANGE IN SIGNS)  (N6 DOT)               DE6FN79
DEC ,315591931              N5 DOT                                       DE6FN80
DEC ,329861111              N4 DOT                                       DE6FN80
DEC ,348611111              N3 DOT                                       DE6FN80
DEC ,375000000              N2 DOT                                       DE6FN80
DEC ,416666666              N1 DOT                                       DE6FN80
DEC ,500000000              N0 DOT                                       DE6FN80
TRA DE6F+0725         TO SET UP ROUTINE  (RUNGE KUTTA SUB)               DE6FN80
SXD DE6F+0707,1                 SAVE INDEX REG,                          DE6FN80
SXD DE6F+0708,2               FROM                                       DE6FN80
SXD DE6F+0709,4               MAIN,PROGRAM (NEW R.K. STARTS)             DE6FN80
CLA 0                      (T+2)=H                                       DE6FN81
STO DE6F+0713                H                                           DE6FN81
FDP DE6F+0719         2,                                                 DE6FN81
STQ DE6F+0714            H/2                                             DE6FN81
STQ DE6F+0715            H/2,                                            DE6FN81
FMP DE6F+0713            H                                               DE6FN81
STO DE6F+0711            (H SQ)/2                                        DE6FN81
FDP DE6F+0720         4,                                                 DE6FN81
```

DE6FN (Cont'd)

```
      STQ DE6F+0712         (H SQ)/8                                DE6FN8180
      CLA DE6F+0713         H                                       DE6FN8190
      FDP DE6F+0722       6.                                        DE6FN8200
      STQ DE6F+0716         H/6                                     DE6FN8210
      LXA DE6F+0710,1            N                                  DE6FN8220
      CLA 0,1                 SAVE YI (T+N+3)=YI (M.S.)             DE6FN8230
      STO 0,1                 (T+8N+3)=YI MOST SIG.                 DE6FN8240
      CLA 0,1                 SAVE YI PRIME (T+2N+3)                DE6FN8250
      STO 0,1                 (T+4N+3)=YI PRIME MOST SIG.           DE6FN8260
      TIX DE6F+0604,1,1                                             DE6FN8270
      LXA DE6F+0723,1               1                               DE6FN8280
      SXD DE6F+0724,1               K=1(+1)4                        DE6FN8290
      LXA DE6F+0710,6            N TO IR2 AND IR4                   DE6FN8300
      TXL DE6F+0629,1,1             TR IF K=1                       DE6FN8310
      TXL DE6F+0649,1,2             TR IF K=2                       DE6FN8320
      TXL DE6F+0629,1,3             TR IF K=3                       DE6FN8330
      LXA DE6F+0710,2            K=4 (N TO IR4)                     DE6FN8340
      CLA 0,2                 T+3N+3=F(X,Y,Y PRIME)                 DE6FN8350
      FAD 0,2                 T+6N+3=CUM. D(YN) PRIME               DE6FN8360
      STO 0,2                 T+6N+3=NEW D(YN) PRIME                DE6FN8370
      TIX DE6F+0616,2,1                                             DE6FN8380
      TXH DE6F+0669,1,3          OUT IF K=4                         DE6FN8390
      TIX DE6F+0615,1,1             ADD 2F(X,Y,PRIME) IF K=2,3      DE6FN8400
      LXD DE6F+0724,1               K                               DE6FN8410
      TXH DE6F+0626,1,2             TR IF K=3                       DE6FN8420
      CLA DE6F+0717         TEST MISSING 1ST DERIV. IN F            DE6FN8430
      TMI DE6F+0628       *=F(X,Y,Y PRIME)                          DE6FN8440
      TSX 0,4                 V (TR OUT FOR F(X,Y,Y PRIME)          DE6FN8450
      LXD DE6F+0724,1                                               DE6FN8460
      TXI DE6F+0610,1,1            INCREASE K BY 1                  DE6FN8470
      CLA 0                   T+1=(T)                               DE6FN8480
      FAD DE6F+0714         H/2                                     DE6FN8490
      STO COMMON+000          T+H/2 (MOST SIG.)                     DE6FN8500
      STQ COMMON+001          LST. SIG.                             DE6FN8510
      CLA COMMON+001                                                DE6FN8520
      FAD DE6F+0718           CUM. LST. SIG.                        DE6FN8530
      FAD COMMON+000                                                DE6FN8540
      STQ DE6F+0718           SAVE LST. SIG. FOR NEXT INTEGRATION   DE6FN8550
      STO 0                   T+1=(T+H/2)                           DE6FN8560
      TIX DE6F+0639,1,1            IR1=1 IF K=1, IR1=2 IF K=3       DE6FN8570
      LDQ DE6F+0713,1            (H SQ)/8 OR (H SQ)/2 (K=1,3)       DE6FN8580
      FMP 0,2                 T+3N+3=F(X,Y,Y PRIME)                 DE6FN8590
      STO COMMON+000                                                DE6FN8600
      LDQ DE6F+0715,1            (H/2) OR (H) K=1 OR 3              DE6FN8610
      FMP 0,2                 T+4N+3=YN PRIME (M.S.)                DE6FN8620
      FAD COMMON+000          (H SQ)/8 X F(X,Y,YP)                  DE6FN8630
      FAD 0,2                 T+8N+3=YN (M.S.)                      DE6FN8640
      STO 0,2                 T+N+3=YN AT (T+H/2) OR (T+H)          DE6FN8650
      TIX DE6F+0639,2,1            N WAS IN IR2                     DE6FN8660
      LXD DE6F+0724,1                RESTORE K TO 1 OR 3            DE6FN8670
      CLA DE6F+0717         TEST MISSING 1ST DERIV. IN F            DE6FN8680
```

DE6FN (Cont'd)

```
TMI DE6F+0656      *=F(X,Y), +=F(X,Y,Y PRIME)                    DE6FN86
LDQ DE6F+0716,1           H/2 OR H                               DE6FN87
FMP 0,4                 T+3N+3=F(X,Y,Y PRIME)                    DE6FN87
FAD 0,4                 T+4N+3=YN PRIME (M.S.)                   DE6FN87
STO 0,4                 T+2N+3=NEW Y PRIME                       DE6FN87
TIX DE6F+0651,4,1                                                DE6FN87
LXA DE6F+0710,2           N                                      DE6FN87
TXL DE6F+0664,1,1           TR TO STORE IF K=1                   DE6FN87
CLA 0,2                 T+3N+3=F(X,Y,Y PRIME)                    DE6FN87
FAD 0,2                 T+7N+3=CUM.DELTA(YN)                     DE6FN87
STO 0,2                 T+7N+3                                   DE6FN87
TIX DE6F+0658,2,1                                                DE6FN88
TXL DE6F+0615,1,2              TR IF K=2 (CONNOT=1)              DE6FN88
TXI DE6F+0615,1,-1             REDUCE K FROM 3 TO 2              DE6FN88
CLA 0,2                 T+3N+3=F(X,Y,Y PRIME)                    DE6FN88
STO 0,2                 T+6N+3=STORE FOR D(YN)PRIME              DE6FN88
STO 0,2                 T+7N+3=STORE FOR D(YN)                   DE6FN88
TIX DE6F+0664,2,1           N WAS IN IR2                         DE6FN88
TRA DE6F+0626      TR FOR K=2                                    DE6FN88
LXA DE6F+0717,4          2N                                      DE6FN88
LXA DE6F+0710,2           N                                      DE6FN88
LDQ DE6F+0716        H/6                                         DE6FN89
FMP 0,4                 T+7N+3=D(YN)P AND D(YN)                  DE6FN89
STQ COMMON+002                                                   DE6FN89
UFA 0,2                 T+4N+3=YN PRIME (M.S.)                   DE6FN89
STO COMMON+000                                                   DE6FN89
STQ COMMON+001                                                   DE6FN89
CLA COMMON+001                                                   DE6FN89
UFA 0,2                 T+5N+3=YN PRIME (LST. SIG.)              DE6FN89
UFA COMMON+002                                                   DE6FN89
FAD COMMON+000                                                   DE6FN89
STO 0,4                 T+3N+3=(YN)P AND (YN) (M.S.)             DE6FN90
STQ 0,4                 T+7N+3=(YN)P AND (YN) (L.S.)             DE6FN90
TNX DE6F+0686,4,1           (YN)  DESTROYS F(X,Y,Y PRIME)        DE6FN90
TIX DE6F+0671,2,1          (YN)P AT T+N+3 (COMPLETED)            DE6FN90
TRA DE6F+0670      (YN)  AT T+2N+3 (NOT COMPLETED)               DE6FN90
LXA DE6F+0710,2           N                                      DE6FN90
LDQ DE6F+0713        H                                           DE6FN90
FMP 0,2                 T+3N+3=(YN PRIME + H/6(DY)               DE6FN90
STQ COMMON+002                                                   DE6FN90
UFA 0,2                 T+8N+3=YN (M.S.)                         DE6FN90
STO COMMON+000                                                   DE6FN91
STQ COMMON+001                                                   DE6FN91
CLA COMMON+001                                                   DE6FN91
UFA 0,2                 T+9N+3=YN (L.S.)                         DE6FN91
UFA COMMON+002                                                   DE6FN91
FAD COMMON+000                                                   DE6FN91
STO 0,2                 T+N+3=Y(N+1) (M.S.)                      DE6FN91
STQ 0,2                 T+9N+3=Y(N+1) (L.S.)                     DE6FN91
CLA 0,2                 T+6N+3=Y(N+1)P (L.S.)                    DE6FN91
STO 0,2                 T+5N+3 (SAVE LST. SIG.)                  DE6FN91
```

DE6FN (Cont'd)

```
TIX DE6F+0687,2,1                                          DE6FN9200
TSX 0,4                  V (FORM 2ND DERIV.)               DE6FN9210
LXD DE6F+0709,4                                            DE6FN9220
LXD DE6F+0707,1                                            DE6FN9230
LXD DE6F+0708,2                                            DE6FN9240
TRA 1,4                 LEAVE SUB. FROM 2ND ENTRY          DE6FN9250
PZE 2,0,0               IR1 SAVED IN DECR.                 DE6FN9260
BSS 11                                                     DE6FN9270
DEC 2.                  2.                                 DE6FN9280
DEC 4.                  4.                                 DE6FN9290
DEC 8.                  8.                                 DE6FN9300
DEC 6.                  6.                                 DE6FN9310
DEC 1                   1                                  DE6FN9320
BSS 1                   K=1(+1)4                           DE6FN9330
SXD DE6F+0707,1              R.K. SET UP ENTRY             DE6FN9340
SXD DE6F+0708,2                                            DE6FN9350
SXD DE6F+0709,4                                            DE6FN9360
STZ DE6F+0718            CLEAR LST. SIG. OF T              DE6FN9370
CLA 1,4                 T,0,V                              DE6FN9380
STA DE6F+0708       T=ADDRESS                              DE6FN9390
STO DE6F+0717          -=F(X,Y), +=F(X,Y,Y PRIME)          DE6FN9400
STA DE6F+0737                                              DE6FN9410
ARS 18                   V TO ADDR.                        DE6FN9420
STA DE6F+0796       V                                      DE6FN9430
STA DE6F+0702        V                                     DE6FN9440
STA DE6F+0626                                              DE6FN9450
CLA                     (T)=N                              DE6FN9460
STO DE6F+0710         N                                    DE6FN9470
CAL DE6F+0708       T                                      DE6FN9480
ADD DE6F+0723       1                                      DE6FN9490
                                                           DE6FN9500
STA DE6F+0629       T+1                                    DE6FN9510
STA DE6F+0637                                              DE6FN9520
ADD DE6F+0723       1                                      DE6FN9530
STA DE6F+0591       T+2                                    DE6FN9540
ADD DE6F+0723       1                                      DE6FN9550
ADM DE6F+0710         N                                    DE6FN9560
STA DE6F+0604       T+N+3=YI (STARTS AT T+3)               DE6FN9570
STA DE6F+0697       T+N+3=YI                               DE6FN9580
STA DE6F+0646       T+N+3                                  DE6FN9590
ADM DE6F+0710         N                                    DE6FN9600
STA DE6F+0606       T+2N+3=YI PRIME (STARTS T+N+3)         DE6FN9610
STA DE6F+0654                                              DE6FN9620
ADM DE6F+0710         N                                    DE6FN9630
STA DE6F+0616       T+3N+3=2ND DERIV.                      DE6FN9640
STA DE6F+0681                                              DE6FN9650
STA DE6F+0640       T+3N+3                                 DE6FN9660
STA DE6F+0652       T+3N+3                                 DE6FN9670
STA DE6F+0658       T+3N+3                                 DE6FN9680
STA DE6F+0664                                              DE6FN9690
STA DE6F+0688       T+3N+3                                 DE6FN9700
```

DE6FN (Cont'd)

```
 ADM DE6F+0710      N                                   DE6FN9710
 STA DE6F+0653    T+4N+3                                DE6FN9720
                                                        DE6FN9730
 STA DE6F+0607                                          DE6FN9740
 STA DE6F+0643    T+4N+3                                DE6FN9750
 STA DE6F+0674    T+4N+3                                DE6FN9760
 ADM DE6F+0710      N                                   DE6FN9770
 STA DE6F+0793    T+5N+3                                DE6FN9780
 STA DE6F+0700                                          DE6FN9790
 STA DE6F+0678                                          DE6FN9800
 ADM DE6F+0710      N                                   DE6FN9810
 STA DE6F+0617    T+6N+3                                DE6FN9820
 STA DE6F+0618                                          DE6FN9830
 STA DE6F+0665                                          DE6FN9840
 STA DE6F+0699    T+6N+3                                DE6FN9850
 ADM DE6F+0710      N                                   DE6FN9860
 STA DE6F+0660    T+7N+3                                DE6FN9870
 STA DE6F+0672                                          DE6FN9880
 STA DE6F+0659                                          DE6FN9890
 STA DE6F+0666                                          DE6FN9900
 STA DE6F+0682    T+7N+3                                DE6FN9910
 ADM DE6F+0710      N                                   DE6FN9920
 STA DE6F+0605    T+8N+3                                DE6FN9930
 STA DE6F+0645                                          DE6FN9940
 STA DE6F+0690                                          DE6FN9950
 ADM DE6F+0710      N                                   DE6FN9960
 STA DE6F+0694    T+9N+3                                DE6FN9970
 STA DE6F+0698    T+9N+3                                DE6FN9980
 STA DE6F+0794                                          DE6FN9990
```

DE6FN (Cont'd)

```
 CAL DE6F+0710          N                                       DE6FN10000
 PAX 0,1                  N TO IR1                              DE6FN10010
 ADM DE6F+0710           N                                      DE6FN10020
 STA DE6F+0717          2N TO ADDR. OF SAVE+10                  DE6FN10030
 STZ 0,1                  T+5N+3 (CLEAR Y PRIME L.S.)           DE6FN10040
 STZ 0,1                  T+9N+3 (CLEAR Y LST. SIG.)            DE6FN10050
 TIX DE6F+0793,1,1                                              DE6FN10060
 TSX 0,4                   V (FORM 2ND DERIV.)                  DE6FN10070
 LXD DE6F+0709,4               RESTORE IRC                      DE6FN10080
 TXI DE6F+0704,4,-1              TRA 2,4 FROM HERE              DE6FN10090
 SXD DE6F+0504,1           COWELL SETUP                         DE6FN10100
 SXD DE6F+0505,2                                                DE6FN10110
 SXD DE6F+0506,4                                                DE6FN10120
 CLA 1,4                  T,0,V                                 DE6FN10130
 STO DE6F+0805        FIX RUNGE KUTTA (1ST ENTRY)               DE6FN10140
 TSX DE6F+0587,4          TR TO SET UP RUNGE KUTTA              DE6FN10150
 PZE                      T,0,V                                 DE6FN10160
 LXD DE6F+0506,4                RESTORE IR4 AFTER R.K.          DE6FN10170
 CAL DE6F+0702         V                                        DE6FN10180
 STA DE6F+0107       V                                          DE6FN10190
 STA DE6F+0230        V                                         DE6FN10200
 STA DE6F+0271       V                                          DE6FN10210
 CAL DE6F+0717           2N                                     DE6FN10220
 STA DE6F+0526                                                  DE6FN10230
 ADM DE6F+0645         T+8N+3                                   DE6FN10240
 STA DE6F+0134        T+10N+3=T+3N+3+( 7N)                      DE6FN10250
 STA DE6F+0066                                                  DE6FN10260
 STA DE6F+0331                                                  DE6FN10270
 ADM DE6F+0710           N                                      DE6FN10280
 STA DE6F+0103        T+11N+3=T+3N+3+(8N)                       DE6FN10290
 STA DE6F+0111        T+3N+3+(8N)                               DE6FN10300
 STA DE6F+0146        T+4N+3+(7N)                               DE6FN10310
 STA DE6F+0068        T+11N+3                                   DE6FN10320
 STA DE6F+0321                                                  DE6FN10330
 CAL DE6F+0794         T+9N+3                                   DE6FN10340
 STA DE6F+0257                                                  DE6FN10350
 CAL DE6F+0700         T+5N+3                                   DE6FN10360
 STA DE6F+0268                                                  DE6FN10370
 CAL DE6F+0681         T+3N+3                                   DE6FN10380
 STA DE6F+0049                                                  DE6FN10390
 STA DE6F+0110                                                  DE6FN10400
 STA DE6F+0239                                                  DE6FN10410
 STA DE6F+0282        T+3N+3                                    DE6FN10420
 CAL DE6F+0606         T+2N+3                                   DE6FN10430
 STA DE6F+0067                                                  DE6FN10440
 STA DE6F+0104                                                  DE6FN10450
 STA DE6F+0223                                                  DE6FN10460
 STA DE6F+0265        T+2N+3                                    DE6FN10470
 CAL DE6F+0697         T+N+3                                    DE6FN10480
 STA DE6F+0065                                                  DE6FN10490
 STA DE6F+0102                                                  DE6FN10500
```

DE6FN (Cont'd)

```
STA DE6F+0129                                   DE6FN105
STA DE6F+0213                                   DE6FN105
STA DE6F+0254                                   DE6FN105
STA DE6F+0340     T+N+3                         DE6FN105
CAL DE6F+0629      T+1                          DE6FN105
STA DE6F+0093                                   DE6FN105
STA DE6F+0094                                   DE6FN105
STA DE6F+0097                                   DE6FN105
STA DE6F+0099                                   DE6FN105
STA DE6F+0193                                   DE6FN106
STA DE6F+0200     T+1                           DE6FN106
CAL DE6F+0591      T+2                          DE6FN106
STA DE6F+0032                                   DE6FN106
STA DE6F+0091                                   DE6FN106
STA DE6F+0923      T+2                          DE6FN106
STA DE6F+0098                                   DE6FN106
STA DE6F+0115                                   DE6FN106
STA DE6F+0192     T+2                           DE6FN106
LDQ DE6F+0538      8                            DE6FN106
MPY DE6F+0710        N                          DE6FN107
STQ DE6F+0528      8N                           DE6FN107
CAL DE6F+0528      8N                           DE6FN107
SBM DE6F+0710        N                          DE6FN107
STA DE6F+0527      7N                           DE6FN107
ADM DE6F+0526      2N                           DE6FN107
ADM DE6F+0526      2N                           DE6FN107
STA DE6F+0529      11N                          DE6FN107
ADM DE6F+0672      T+7N+3                       DE6FN107
STA DE6F+0050     T+18N+3=T+11N+3+(7N)          DE6FN107
STA DE6F+0317     T+18N+3=T+11N+3+(7N)          DE6FN108
ADM DE6F+0710        N                          DE6FN108
STA DE6F+0101     T+19N+3=T+11N+3+(8N)          DE6FN108
ADM DE6F+0537      7                            DE6FN108
STA DE6F+0530      T+19N+10=T+19N+3+(7)         DE6FN108
CAL DE6F+0101     T+19N+3                       DE6FN108
ADM DE6F+0529      11N                          DE6FN108
STA DE6F+0178     T+30N+3=T+19N+3+(11N)         DE6FN108
STA DE6F+0234                                   DE6FN108
STA DE6F+0278                                   DE6FN108
STA DE6F+0309     T+30N+3                       DE6FN109
ADM DE6F+0537      7                            DE6FN109
STA DE6F+0210     T+30N+10=T+19N+3+(11N+7)      DE6FN109
ADM DE6F+0532      2                            DE6FN109
STA DE6F+0220     T+30N+12=T+19N+3+(11N+9)      DE6FN109
CAL DE6F+0317     7+18N+3                       DE6FN109
SBM DE6F+0526      2N                           DE6FN109
STA DE6F+0149     T+16N+3                       DE6FN109
SBM DE6F+0710        N                          DE6FN109
STA DE6F+0138     T+15N+3                       DE6FN109
LXA DE6F+0710,1            N                    DE6FN110
SXD DE6F+0389,1               N TO DECR.        DE6FN110
```

DE6FN (Cont'd)

```
CLA 2,4                 PZE=VAR. STEP SIZE (B,0,R)                 DE6FN11020
STO DE6F+0501          MZE=FIXED STEP SIZE                         DE6FN11030
STZ COMMON+000                                                     DE6FN11040
STA COMMON+000            B (B LESS THAN OR =N)                    DE6FN11050
STZ DE6F+0516                                                      DE6FN11060
STD DE6F+0516          R=TOTAL R.K. LOOPS PER D(T)                 DE6FN11070
CLA DE6F+0516                                                      DE6FN11080
ARS 18                                                             DE6FN11090
TNZ DE6F+0903                                                      DE6FN11100
LXA DE6F+0534,1                 4                                  DE6FN11110
SXD DE6F+0516,1                    IF R=0, SET R=4                 DE6FN11120
CLA DE6F+0534          4                                           DE6FN11130
ADD DE6F+0524          233000000000                                DE6FN11140
FAD DE6F+0571          FLOAT FIXED PT. NO.                         DE6FN11150
STO DE6F+0517          R IN FLOATING PT.                           DE6FN11160
CLA COMMON+000            TEST B=0                                 DE6FN11170
TNZ DE6F+0910                                                      DE6FN11180
CLA DE6F+0710            IF B=0, SET B=N                           DE6FN11190
STA COMMON+000              N                                      DE6FN11200
CAL DE6F+0710            N                                         DE6FN11210
SBM COMMON+000              B (OR N, IF B=0)                       DE6FN11220
TPL DE6F+0914          PROGRAMMER GOOFED, IF MINUS                 DE6FN11230
CLM                       SET B=N IF B TOO LARGE (N-B=0)           DE6FN11240
ALS 18                    N-B TO DECR.                             DE6FN11250
STD DE6F+0338         1ST B EQNS. TESTED                           DE6FN11260
STD DE6F+0312         N-B                                          DE6FN11270
CLA 3,4                   S                                        DE6FN11280
TNZ DE6F+0920          IF S=0, SET S=9                             DE6FN11290
CLA DE6F+0954           1E-9                                       DE6FN11300
STO COMMON+000              S (SEE VMIN,VMAX,WMAX.                 DE6FN11310
STZ DE6F+0503         +=R.K., -=COWELL                             DE6FN11320
STZ DE6F+0709            +=1ST R.K. INTEGRATION, -=2ND, ETC.       DE6FN11330
CLA                       T+2=DELTA(T)=H                           DE6FN11340
STO DE6F+0519                                                      DE6FN11350
CLA 6,4                     A=MIN. Y ALLOWED                       DE6FN11360
TNZ DE6F+0928                                                      DE6FN11370
CLA DE6F+0543          1, (IF A=0, SET A=1.)                       DE6FN11380
STO DE6F+0511                                                      DE6FN11390
CLA 4,4                    MINIMUM H ALLOWED  (FLOATING PT.)       DE6FN11400
SSP                                                                DE6FN11410
STO DE6F+0509                                                      DE6FN11420
CLA 5,4                    MAXIMUM H ALLOWED (FLOATING PT.)        DE6FN11430
SSP                                                                DE6FN11440
TNZ DE6F+0936           IF H=0, SET HMAX= 1E18                     DE6FN11450
CLA DE6F+0525           1E18                                       DE6FN11460
STO DE6F+0510                                                      DE6FN11470
LDQ DE6F+0519          DELTA(T)=H                                  DE6FN11480
FMP DE6F+0519                                                      DE6FN11490
STO DE6F+0520          H SQ.                                       DE6FN11500
CLA COMMON+000             1E-S                                    DE6FN11510
FDP DE6F+0520           H SQ.                                      DE6FN11520
```

DE6FN (Cont'd)

```
         FMP DE6F+0952       .1                                            DE6FN115
         STO DE6F+0512       VMIN=(10)**(-1-S)/H SQ.                       DE6FN115
         STO DE6F+0514       WMAX                                          DE6FN115
         FDP DE6F+0953       1E-4                                          DE6FN115
         STQ DE6F+0513       VMAX=(10)**(3-S)/H SQ.                        DE6FN115
         STZ DE6F+0500       SWITCH (PROGRAMMER SETS NON/ZERO)             DE6FN115
         CLA DE6F+0538       8 (SET UP COUNTER IN CASE                     DE6FN115
         STO DE6F+0502       OF CHANGE IN COORD.)                          DE6FN116
         LXD DE6F+0506,4         RETURN TO PROGRAMMER. (1ST ENTRY)         DE6FN116
         TXI DE6F+0074,4,-6      TRA 7,4                                   DE6FN116
         DEC .1                  .1                                        DE6FN116
         DEC 1E-4                .0001=1E-4                                DE6FN116
         DEC 1E-9                1E-9 (IF S=0)                             DE6FN116
BREAK    EQU *                                                             DE6FN116
END      EQU *                                                             DE6FN116
```

Page Intentionally Left Blank

Function: DOT

Purpose: To find the dot product of two three-dimensional vectors.

Calling Sequence:

Z = DOT (X, Y)

Input and **Output**

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	X	3			First input vector
I	Y	3			Second input vector
O	Z				Z = X(1)Y(1)+X(2)Y(2)+X(3)Y(3)

Common storages used or required: None

Subroutines required: None

Functions required: None

Approximate number of storages required:

DOT

```
*       LABEL                                                    DOT0
*       SYMBOL TABLE                                             DOT0
        FUNCTION DOT(X,Y)                                        DOT0000
        DIMENSION X(3),Y(3)                                      DOT0010
        DOT = X(1)*Y(1) + X(2)*Y(2) + X(3)*Y(3)                  DOT0020
        RETURN                                                   DOT0030
        END                                                      DOT0
```

Subroutine: EARTR

Purpose: Subroutine EARTR determines which tracking stations can observe the vehicle at the time EARTR is called. The subroutine then updates the state covariance matrix for the types of observations being performed by the tracking station. The possible types of observations are: range, range rate, azimuth-elevation, and direction cosines. The measurements may be corrupted by the following types of errors: random, bias, time bias, and station location errors. The subroutine input and output quantities are in common storage.

Calling Sequence:

CALL EARTR

Input and **Output**

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I					Quantities obtained from common
O					Quantities placed in common

Common storages used or required: T, S, C, IC

Subroutines required: CORRTP, DOT, GHA, INTRI, MATRX, NUTAIT, SETN, TRAC, TRANSH

Functions required: ARKTNS, ATAN, COS, FNORM, SQRT, SIN

Approximate number of storages required: 1530

EARTR Partial Derivatives

To perform the covariance matrix updating for earth based tracking observations, a number of partial derivatives are required. The updating of the covaniance matrix for observations requires the partial derivatives relating the types of measurements and the vehicle state. The evaluation of the errors in the measurements due to station location errors and time bias requires the partials which relate the measurements and the station latitude, longitude, altitude, and time.

The following quantities and relationships are used in the derivations of the various partials. Four important vectors are obtained by calling subroutine TRAC. Three unit vectors, $\hat{U}$, $\hat{E}$, $\hat{N}$, which form a topocentric orthognal coordinate system and the vector X_T are obtained from the center of the earth to tracker.

The unit vectors are the up, east, and north vectors. These vectors may be written as follows:

$U_1 = \cos(LAT)\ \cos(LON)$

$U_2 = \cos(LAT)\ \sin(LON)$

$U_3 = \sin(LAT)$

$E_1 = -\sin(LON)$

$E_2 = \cos(LON)$

$E_3 = 0$

$N_1 = -\sin(LAT)\ \cos(LON)$

$N_2 = -\sin(LAT)\ \sin(LON)$

$N_3 = \cos(LAT)$

$LON = \omega(t + t_o) + LON'$

LON' = STATION LONGITUDE

ω = EARTH'S ROTATION RATE

ωt_o = GREENWICH'S LONGITUDE AT EPOCH

$$X_T = \left(\frac{R_E}{C} + H \right) U_1$$

$$Y_T = \left(\frac{R_E}{C} + H \right) U_2$$

$$Z_T = \left(\frac{R_P^2}{R_E C} + H \right) U_3$$

H = ALTITUDE

$$C = \left[\cos^2(\text{LAT}) + \frac{R_P^2}{R_E^2} \sin^2(\text{LAT}) \right]^{1/2}$$

R_E = EQUATORIAL RADIUS

R_P = POLAR RADIUS

The vehicle's position, $\vec{X}_v$, and velocity, $\dot{\vec{X}}_v$, are obtained from common storage.

$$\vec{X}_v = \begin{pmatrix} X_v \\ Y_v \\ Z_v \end{pmatrix} \qquad \dot{\vec{X}}_v = \begin{pmatrix} \dot{X}_v \\ \dot{Y}_v \\ \dot{Z}_v \end{pmatrix}$$

The tracking station velocity is obtained by differentiating $\vec{X}_T$.

$$\frac{d}{dt} (\vec{X}_T) = \dot{\vec{X}}_T = \begin{pmatrix} \dot{X}_T \\ \dot{Y}_T \\ \dot{Z}_T \end{pmatrix}$$

$$\dot{X}_T = -\omega \left(\frac{R_E}{C} + H \right) \cos(\text{LAT}) \sin(\text{LON})$$

$$\dot{Y}_T = \omega \left(\frac{R_E}{C} + H \right) \cos(\text{LAT}) \cos(\text{LON})$$

$$\dot{Z}_T = 0$$

The vector from the tracking station to vehicle may then be written as

$$\underset{3\times 1}{\vec{X}} = \underset{3\times 1}{\begin{pmatrix} X_1 \\ X_2 \\ X_3 \end{pmatrix}} = \underset{3\times 1}{\vec{X}_v - \vec{X}_T} = \underset{3\times 1}{\begin{pmatrix} X_v - X_T \\ Y_v - Y_T \\ Z_v - Z_T \end{pmatrix}}$$

A relationship which will be used in the derivations is

$$\frac{\partial PAR}{\partial \vec{X}_v} = \frac{\partial PAR}{\partial \vec{X}}$$

The FORTRAN program names for the above vectors are as follows:

FORTRAN NAME	DIMENSION	DERIVATION NAME	DIMENSION
U	(3)	U	(3)
E	(3)	E	(3)
EN	(3)	N	(3)
X	(3)	X_v	(3)
RT	(3)	X_T	(3)
Y	(3)	$X = X_v - X_T$	(3)
A	(1)	R_E	(1)
B	(1)	R_P	(1)
RAT	(1)	$\lvert X \rvert$	(1)
XD	(3)	$\dot{X} = \dot{X}_v - \dot{X}_T$	(3)

The quantities being measured are the following:

$$R = \text{RANGE} = |\vec{X}| = (\vec{X}\cdot\vec{X})^{1/2}$$

$$\dot{R} = \text{RANGE RATE} = \frac{d}{dt}|\vec{X}| = \frac{\vec{X}\cdot\dot{\vec{X}}}{R}$$

$$AZ = \tan^{-1}\left\{\frac{\vec{X}\cdot\hat{E}}{\vec{X}\cdot\hat{N}}\right\} \qquad EL = \sin^{-1}\frac{(\hat{U}\cdot\vec{X})}{R}$$

Below is a list of partial derivatives which will be used in the application of the chain rule for partial derivatives.

$$\frac{\partial U_1}{\partial LAT} = N_1 \qquad \frac{\partial E_1}{\partial LAT} = 0$$

$$\frac{\partial U_2}{\partial LAT} = N_2 \qquad \frac{\partial E_2}{\partial LAT} = 0$$

$$\frac{\partial U_3}{\partial LAT} = N_3 \qquad \frac{\partial E_e}{\partial LAT} = 0$$

$$\frac{\partial U_1}{\partial LON} = \cos(LAT)\ \sin(LON) \qquad \frac{\partial E_1}{\partial LON} = -\cos(LON)$$

$$\frac{\partial U_2}{\partial LON} = \cos(LAT)\ \cos(LON) \qquad \frac{\partial E_2}{\partial LON} = -\sin(LON)$$

$$\frac{\partial U_3}{\partial LON} = 0 \qquad \frac{\partial E_3}{\partial LON} = 0$$

$$\frac{\partial U_1}{\partial ALT} = 0 \qquad \frac{\partial E_1}{\partial ALT} = 0$$

$$\frac{\partial U_2}{\partial ALT} = 0 \qquad \frac{\partial E_2}{\partial ALT} = 0$$

$$\frac{\partial U_3}{\partial ALT} = 0 \qquad \frac{\partial E_3}{\partial ALT} = 0$$

$$\frac{\partial N_1}{\partial LAT} = -U_1 \qquad \frac{\partial U_1}{\partial t} = -\omega \sin(LON) \cos(LAT)$$

$$\frac{\partial N_2}{\partial LAT} = -U_2 \qquad \frac{\partial U_2}{\partial t} = \omega \cos(LON) \cos(LAT)$$

$$\frac{\partial N_3}{\partial LAT} = -U_3 \qquad \frac{\partial U_3}{\partial t} = 0$$

$$\frac{\partial N_1}{\partial LON} = \sin(LAT) \sin(LON) \qquad \frac{\partial E_1}{\partial t} = -\omega \cos(LON)$$

$$\frac{\partial N_2}{\partial LON} = -\sin(LAT) \cos(LON) \qquad \frac{\partial E_2}{\partial t} = -\omega \sin(LON)$$

$$\frac{\partial N_3}{\partial LON} = 0 \qquad \frac{\partial E_3}{\partial t} = 0$$

$$\frac{\partial N_1}{\partial ALT} = 0 \qquad \frac{\partial N_1}{\partial t} = \omega \sin(LAT) \sin(LON)$$

$$\frac{\partial N_2}{\partial ALT} = 0 \qquad \frac{\partial N_2}{\partial t} = -\omega \sin(LAT) \cos(LON)$$

$$\frac{\partial N_3}{\partial ALT} = 0 \qquad \frac{\partial N_3}{\partial t} = 0$$

FORTRAN 3x3 matrix called DRT

$$\underset{3\text{x}3}{DRT} = \underset{3\text{x}3}{\begin{pmatrix} \frac{\partial X_1}{\partial LAT} & \frac{\partial X_1}{\partial LON} & \frac{\partial X_1}{\partial ALT} \\ \frac{\partial X_2}{\partial LAT} & \frac{\partial X_2}{\partial LON} & \frac{\partial X_2}{\partial ALT} \\ \frac{\partial X_3}{\partial LAT} & \frac{\partial X_3}{\partial LON} & \frac{\partial X_3}{\partial ALT} \end{pmatrix}}$$

$$DRT(1,1) = -\frac{\partial X_T}{\partial LAT} = \left(\frac{R_E}{C} + H\right) \sin(LAT)\ \cos(LON) + U_1 \frac{R_E}{C^2} \frac{\partial C}{\partial LAT}$$

$$\frac{\partial C}{\partial LAT} = \frac{\sin(LAT)\ \cos(LAT)}{C} \left(1 - \frac{R_P^2}{R_E^2}\right)$$

$$DRT(2,1) = -\frac{\partial Y_T}{\partial LAT} = U_2 \frac{R_E}{C^2} \frac{\partial C}{\partial LAT} + \left(\frac{R_E}{C} + H\right) \sin(LAT)\ \sin(LON)$$

$$DRT(3,1) = -\frac{\partial Z_T}{\partial LAT} = -\left(\frac{R_P^2}{R_E C} + H\right) \cos(LAT) + U_3 \frac{R_P^2}{R_E C^2} \frac{\partial C}{\partial LAT}$$

$$DRT(1,2) = -\frac{\partial X_T}{\partial LON} = -E_1 \left(\frac{R_E}{C} + H\right) \cos(LAT)$$

$$DRT(2,2) = -\frac{\partial Y_T}{\partial LON} = -E_2 \left(\frac{R_E}{C} + H\right) \cos(LAT)$$

$$DRT(3,2) = 0$$

$$DRT(1,3) = -U_1$$

$$DRT(2,3) = -U_2$$

$$DRT(3,3) = -U_3$$

$$\frac{\partial \dot{X}_1}{\partial LAT} = -\frac{\partial \dot{X}_T}{\partial LAT} = -\omega \left(\frac{R_E}{C} + H\right) \sin(LAT)\ \sin(LON)$$

$$\frac{\partial \dot{X}_2}{\partial LAT} = -\frac{\partial \dot{Y}_T}{\partial LAT} = \omega \left(\frac{R_E}{C} + H\right) \sin(LAT)\ \cos(LON)$$

$$\frac{\partial \dot{X}_3}{\partial LAT} = -\frac{\partial \dot{Z}_T}{\partial LAT} = 0$$

$$\frac{\partial \dot{X}_1}{\partial LON} = -\frac{\partial \dot{X}_T}{\partial LON} = \omega\left(\frac{R_E}{C} + H\right) \cos(LAT)\ \cos(LON)$$

$$\frac{\partial \dot{X}_2}{\partial LON} = -\frac{\partial \dot{Y}_T}{\partial LON} = \omega\left(\frac{R_E}{C} + H\right) \cos(LAT)\ \sin(LON)$$

$$\frac{\partial \dot{X}_3}{\partial LON} = -\frac{\partial \dot{Z}_T}{\partial LON} = 0$$

$$\frac{\partial \dot{X}_1}{\partial ALT} = -\frac{\partial \dot{X}_T}{\partial ALT} = \omega\ \cos(LAT)\ \sin(LON)$$

$$\frac{\partial \dot{X}_2}{\partial ALT} = -\frac{\partial \dot{Y}_T}{\partial ALT} = -\omega\ \cos(LAT)\ \cos(LON)$$

$$\frac{\partial X_3}{\partial ALT} = -\frac{\partial Z_T}{\partial ALT} = 0$$

A. The following is a derivation of the partials relating the types of measurements and the vehicle state. In the program, these partials are used as a row vector. The FORTRAN name of the vector is H. H may be written as follows:

$$\underset{1x6}{H} = \underset{1x6}{\left(\frac{\partial MEAS}{\partial X_v}; \frac{\partial MEAS}{\partial Y_v}; \frac{\partial MEAS}{\partial Z_v}; \frac{\partial MEAS}{\partial \dot{X}_v}; \frac{\partial MEAS}{\partial \dot{Y}_v}; \frac{\partial MEAS}{\partial \dot{Z}_v}\right)}$$

(1) Derivation of partials for range measurement

$$R = (\vec{X}\cdot\vec{X})^{1/2}$$

$$\underset{1x6}{H_R} = \underset{1x6}{\left(\frac{\partial R}{\partial \vec{X}_v}; \frac{\partial R}{\partial \dot{\vec{X}}_v}\right)} = \underset{1x6}{\left(\frac{X_1}{R}; \frac{X_2}{R}; \frac{X_3}{R}; 0; 0; 0\right)}$$

(2) Derivation of partials for range rate measurement.

$$\dot{R} = \frac{\vec{X}\cdot\dot{\vec{X}}}{R}$$

$$H_{RR} = \left(\frac{\partial \dot{R}}{\partial \vec{X}_v};\ \frac{\partial \dot{R}}{\partial \dot{\vec{X}}_v}\right) = \left(\frac{\dot{X}_1}{R} - \frac{\dot{R}}{R^2}(X_1);\ \frac{\dot{X}_2}{R} - \frac{\dot{R}}{R^2}(X_2);\ \frac{\dot{X}_3}{R} - \frac{\dot{R}}{R^2}(X_3);\ \frac{X_1}{R};\ \frac{X_2}{R};\ \frac{X_3}{R}\right)$$

(3) Derivation of partials for azimuth measurement.

$$AZ = \mathrm{Tan}^{-1}\left(\frac{\vec{X}\cdot\hat{E}}{\vec{X}\cdot\hat{N}}\right) \equiv \mathrm{Tan}^{-1}\,\gamma$$

$$\frac{\partial AZ}{\partial PAR} = \left[1 + \gamma^2\right]^{-1} \frac{\partial \gamma}{\partial PAR}$$

$$\left[1 + \gamma^2\right]^{-1} = \frac{(\vec{X}\cdot\hat{N})^2}{S^2} \qquad S^2 = (\vec{X}\cdot\hat{E}) + (\vec{X}\cdot\hat{N})^2$$

$$H_{AZ} = \left(\frac{\partial AZ}{\partial \vec{X}_v};\ \frac{\partial AZ}{\partial \dot{\vec{X}}_v}\right) = \left(\frac{E_1(\hat{N}\cdot\vec{X})}{S^2} - \frac{\hat{E}\cdot\vec{X}(N_1)}{S^2};\ \frac{E_2(\hat{N}\cdot\vec{X})}{S^2} - \frac{\hat{E}\cdot\vec{X}(N_2)}{S^2};\ \frac{E_3(\hat{N}\cdot\vec{X})}{S^2} - \frac{\hat{E}\cdot\vec{X}(N_3)}{S^2};\ 0;\ 0;\ 0\right)$$

(4) Derivation of partials for elevation measurement.

$$EL = \mathrm{Sin}^{-1}\left(\frac{\hat{U}\cdot\vec{X}}{R}\right) \equiv \mathrm{Sin}^{-1}\,\gamma$$

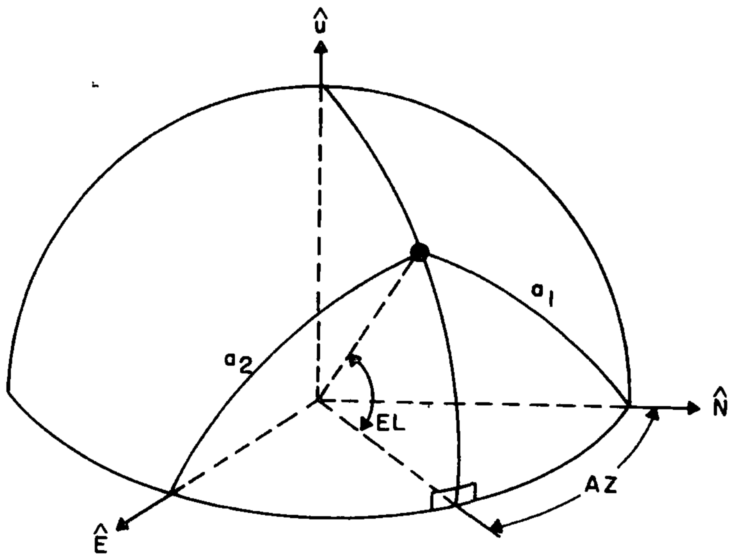
$$\frac{\partial EL}{\partial PAR} = \left[1 - \gamma^2\right]^{-1/2} \frac{\partial \gamma}{\partial PAR}$$

$$(1 - \gamma^2)^{-1/2} = \frac{R}{S}$$

$$H_{EL} = \left(\frac{\partial EL}{\partial \vec{X}} ; \frac{\partial EL}{\partial \dot{\vec{X}}}\right) = \left(\frac{U_1}{S} - \frac{(\hat{U}\cdot\vec{X})}{R^2 S} X_1 ; \frac{U_2}{S} - \frac{(\hat{U}\cdot\vec{X})}{R^2 S} X_2 ;\right.$$

$$\left. \frac{U_3}{S} - \frac{(\hat{U}\cdot\vec{X})}{R^2 S} ; 0; 0; 0 \right)$$

(5) Derivation of partials for direction cosines. The partials for the ℓ and m direction cosines are obtained by combining the azimuth and elevation partials as follows:



$$m = \cos(a_1) = \cos AZ \cos EL$$

$$\ell = \cos(a_2) = \sin AZ \cos EL$$

$$\frac{\partial m}{\partial PAR} = - \cos AZ \sin EL \frac{\partial EL}{\partial PAR} - \sin AZ \cos EL \frac{\partial AZ}{\partial PAR}$$

$$\frac{\partial \ell}{\partial PAR} = - \sin AZ \sin EL \frac{\partial EL}{\partial PAR} + \cos AZ \cos EL \frac{\partial AZ}{\partial PAR}$$

$$H_m = \left(\frac{\partial m}{\partial \bar{x}} \; ; \; \frac{\partial m}{\partial \dot{\bar{x}}} \right) = \left(-\cos AZ \; \sin EL \; -\sin AZ \; \cos EL \right) \begin{pmatrix} H_{AZ} \\ H_{EL} \end{pmatrix}$$

1x6 1x6 1x2 2x6

$$H_\ell = \left(\frac{\partial \ell}{\partial \bar{x}} \; ; \; \frac{\partial \ell}{\partial \dot{\bar{x}}} \right) = \left(-\sin AZ \; \sin EL \quad \cos AZ \; \cos EL \right) \begin{pmatrix} H_{AZ} \\ H_{EL} \end{pmatrix}$$

1x6 1x6 1x2 2x6

B. In order to include errors in the measurements being made due to station location errors and time bias, the partials of the measurements with respect to latitude, longitude, altitude, and time are required. In the program, the station location partials are used as a row vector with FORTRAN name SPART. SPART may be written as follows:

$$SPART = \left(\frac{\partial MEAS}{\partial LAT} \; ; \; \frac{\partial MEAS}{\partial LON} \; ; \; \frac{\partial MEAS}{\partial ALT} \right)$$

(1) Derivations of range measurement error partials.

$$\frac{\partial R}{\partial LAT} = \sum_{i=1}^{3} \frac{\partial R}{\partial x_i} \frac{\partial x_i}{\partial LAT}$$

$$\frac{\partial R}{\partial LAT} = \sum_{i=1}^{3} H_R(1,i) \; DRT(i,1)$$

Similarily

$$\frac{\partial R}{\partial LON} = \sum_{i=1}^{3} H_R(1,i) \; DRT(i,2)$$

$$\frac{\partial R}{\partial ALT} = \sum_{i=1}^{3} H_R(1,i)\ DRT(i,3)$$

$$SPART_R = \left(\frac{\partial R}{\partial LAT};\ \frac{\partial R}{\partial LON};\ \frac{\partial R}{\partial ALT}\right)$$

(2) Derivation of range rate measurement error partials.

$$\frac{\partial \dot{R}}{\partial LAT} = \sum_{i=1}^{3} \frac{\partial \dot{R}}{\partial X_i}\ \frac{\partial X_i}{\partial LAT} + \frac{\partial \dot{R}}{\partial \dot{X}_i}\ \frac{\partial \dot{X}_i}{\partial LAT}$$

$$\frac{\partial \dot{R}}{\partial LAT} = \sum_{i=1}^{3} H_{RR}(1,i)\ DRT(i,1) + H_{RR}(1,i+3)\ \frac{\partial \dot{X}_i}{\partial LAT}$$

Similarly

$$\frac{\partial \dot{R}}{\partial LON} = \sum_{i=1}^{3} H_{RR}(1,i)\ DRT(i,2) + H_{RR}(1,i+3)\ \frac{\partial \dot{X}_i}{\partial LON}$$

$$\frac{\partial \dot{R}}{\partial ALT} = \sum_{i=1}^{3} H_{RR}(1,i)\ DRT(i,3) + H_{RR}(1,i+3)\ \frac{\partial \dot{X}_i}{\partial ALT}$$

$$SPART_{RR} = \left(\frac{\partial \dot{R}}{\partial LAT};\ \frac{\partial \dot{R}}{\partial LON};\ \frac{\partial \dot{R}}{\partial ALT}\right)$$

(3) Derivation of azimuth measurement error partials.

$$\frac{\partial AZ}{\partial LAT} = \sum_{i=1}^{3} \frac{\partial AZ}{\partial X_i}\ \frac{\partial X_i}{\partial LAT} + \overbrace{\cancel{\frac{\partial AZ}{\partial E_i}\ \frac{\partial E_i}{\partial LAT}}}^{(=0)} + \frac{\partial AZ}{\partial N_i}\ \frac{\partial N_i}{\partial LAT}$$

$$\frac{\partial AZ}{\partial N_i} = -\ \frac{X_i(\vec{X}\cdot\hat{E})}{S^2} \qquad i = 1,2,3$$

$$\frac{\partial AZ}{\partial LAT} = \sum_{i=1}^{3} H_{AZ}(1,i)\ DRT(i,1) - \frac{X_i(\vec{X}\cdot\hat{E})}{S^2}\ \frac{\partial N_i}{\partial LAT}$$

$$\frac{\partial AZ}{\partial LON} = \sum_{i=1}^{3} \frac{\partial AZ}{\partial X_i}\ \frac{\partial X_i}{\partial LON} + \frac{\partial AZ}{\partial E_i}\ \frac{\partial E_i}{\partial LON} + \frac{\partial AZ}{\partial N_i}\ \frac{\partial N_i}{\partial LON}$$

$$\frac{\partial AZ}{\partial E_i} = \frac{X_i(\vec{X}\cdot\hat{N})}{S^2} \qquad i = 1,2,3$$

$$\frac{\partial AZ}{\partial LON} = \sum_{i=1}^{3} H_{AZ}(1,i)\ DRT(i,2) + \frac{X_i}{S^2}(\vec{X}\cdot\hat{N})\ \frac{\partial E_i}{\partial LON} - \frac{X_i}{S^2}(\vec{X}\cdot\hat{E})\ \frac{\partial N_i}{\partial LON}$$

$$\frac{\partial AZ}{\partial ALT} = \sum_{i=1}^{3} \frac{\partial AZ}{\partial X_i}\ \frac{\partial X_i}{\partial ALT} + \cancel{\frac{\partial AZ}{\partial E_i}\ \overset{(=0)}{\frac{\partial E_i}{\partial ALT}}} + \cancel{\frac{\partial AZ}{\partial N_i}\ \overset{(=0)}{\frac{\partial N_i}{\partial ALT}}}$$

$$\frac{\partial AZ}{\partial ALT} = \sum_{i=1}^{3} H_{AZ}(1,i)\ DRT(i,3)$$

$$SPART_{AZ} = \left(\frac{\partial AZ}{\partial LAT}\ ;\ \frac{\partial AZ}{\partial LON}\ ;\ \frac{\partial AZ}{\partial ALT} \right)$$

(4) Derivation of elevation measurement error partials.

$$\frac{\partial EL}{\partial LAT} = \sum_{i=1}^{3} \frac{\partial EL}{\partial X_i}\ \frac{\partial X_i}{\partial LAT} + \frac{\partial EL}{\partial U_i}\ \frac{\partial U_i}{\partial LAT}$$

$$\frac{\partial EL}{\partial U_i} = \frac{X_i}{S} \qquad i = 1,2,3$$

$$\frac{\partial EL}{\partial LAT} = \sum_{i=1}^{3} H_{EL}(1,i)\ DRT(i,1) + \frac{X_i}{S}\ \frac{\partial U_i}{\partial LAT}$$

$$\frac{\partial EL}{\partial LON} = \sum_{i=1}^{3} \frac{\partial EL}{\partial X_i}\ \frac{\partial X_i}{\partial LON} + \frac{\partial EL}{\partial U_i}\ \frac{\partial U_i}{\partial LON}$$

$$\frac{\partial EL}{\partial LON} = \sum_{i=1}^{3} H_{EL}(1,i)\ DRT(i,2) + \frac{X_i}{S}\ \frac{\partial U_i}{\partial LON}$$

$$\frac{\partial EL}{\partial ALT} = \sum_{i=1}^{3} \frac{\partial EL}{\partial X_i}\ \frac{\partial X_i}{\partial ALT} + \overset{(=0)}{\cancel{\frac{\partial EL}{\partial U_i}\ \frac{\partial U_i}{\partial ALT}}}$$

$$\frac{\partial EL}{\partial ALT} = \sum_{i=1}^{3} H_{ALT}(1,i)\ DRT(i,3)$$

$$SPART_{EL} = \left(\frac{\partial EL}{\partial LAT}\ ;\ \frac{\partial EL}{\partial LON}\ ;\ \frac{\partial EL}{\partial ALT} \right)$$

(5) Derivation of direction cosine measurement error partials.

$$\underset{1x3}{SPART_m} = \underset{1x2}{\left(-\cos AZ\ \sin EL\ \ -\ \sin AZ\ \cos EL \right)} \underset{2x3}{\begin{pmatrix} SPART_{AZ} \\ SPART_{EL} \end{pmatrix}}$$

$$\underset{1x3}{SPART_\ell} = \underset{1x2}{\left(-\sin AZ\ \sin EL\quad \cos AZ\ \cos EL \right)} \underset{2x3}{\begin{pmatrix} SPART_{AZ} \\ SPART_{EL} \end{pmatrix}}$$

(6) Derivations of time derivatives of measurement quantities to permit inclusion of time bias errors.

$$\frac{\partial R}{\partial t} = \dot{R} = \frac{\vec{X}\cdot\dot{\vec{X}}}{R}$$

$$\frac{\partial AZ}{\partial t} = \sum_{i=1}^{3} \frac{\partial AZ}{\partial X_i}\frac{\partial X_i}{\partial t} + \frac{\partial AZ}{\partial E_i}\frac{\partial E_i}{\partial t} + \frac{\partial AZ}{\partial N_i}\frac{\partial N_i}{\partial t}$$

$$\frac{\partial AZ}{\partial t} = \sum_{i=1}^{3} H_{AZ}(1,i)\,\dot{X}_i + \frac{\partial AZ}{\partial E_i}\dot{E}_i + \frac{\partial AZ}{\partial N_i}\dot{N}_i$$

$$\frac{\partial EL}{\partial t} = \sum_{i=1}^{3} H_{EL}(1,i)\,\dot{X}_i + \frac{\partial EL}{\partial E_i}\dot{E}_i + \frac{\partial EL}{\partial N_i}\dot{N}_i$$

(7) The time partials for the ℓ and m direction cosines are obtained by the following combination of azimuth and elevation partials.

$$\frac{\partial m}{\partial t} = \begin{pmatrix} -\cos AZ\ \sin EL & -\sin AZ\ \cos EL \end{pmatrix} \begin{pmatrix} \frac{\partial AZ}{\partial t} \\ \frac{\partial EL}{\partial t} \end{pmatrix}$$

1x1 1x2 2x1

$$\frac{\partial}{\partial t} = \begin{pmatrix} -\sin AZ\ \sin EL & \cos AZ\ \cos EL \end{pmatrix} \begin{pmatrix} \frac{\partial AZ}{\partial t} \\ \frac{\partial EL}{\partial t} \end{pmatrix}$$

1x1 1x2 2x1

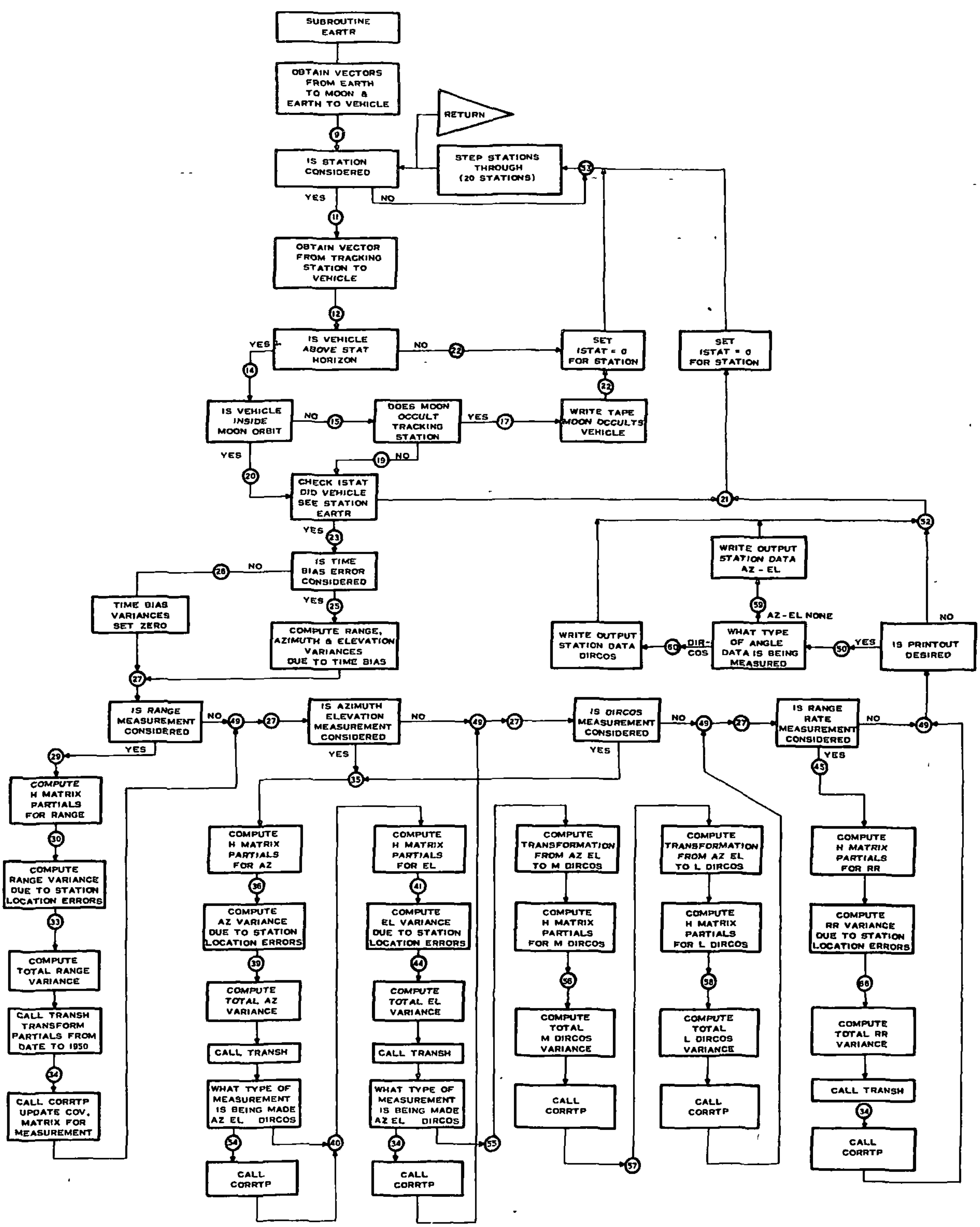
The errors in the measurements are computed in the following manner for station location errors and time bias.

$$QQS = \left(SPART_{MEAS}\right)^2 \begin{pmatrix} \sigma^2 LAT \\ \sigma^2 LON \\ \sigma^2 ALT \end{pmatrix} \qquad \text{Station Location Error}$$

$$\begin{matrix} 1x1 & & 1x3 & & 3x1 \end{matrix}$$

$$QQDOT = \left\{\frac{\partial MEAS}{\partial t} \cdot \sigma_{TIME}\right\}^2 \qquad \text{Time Bias Error}$$

The computation of the H matrices and the measurement errors are the primary computations in EARTR. These quantities are input to subroutine CORRTP which does the weighting of the measurement and updating of the state covariance matrix.


SUBROUTINE EARTR
OBTAIN VECTORS FROM EARTH TO MOON & EARTH TO VEHICLE
RETURN
IS STATION CONSIDERED
STEP STATIONS THROUGH (20 STATIONS)
YES
NO
OBTAIN VECTOR FROM TRACKING STATION TO VEHICLE
IS VEHICLE ABOVE STAT HORIZON
SET ISTAT = 0 FOR STATION
SET ISTAT = 0 FOR STATION
IS VEHICLE INSIDE MOON ORBIT
DOES MOON OCCULT TRACKING STATION
WRITE TAPE MOON OCCULTS VEHICLE
CHECK ISTAT DID VEHICLE SEE STATION EARTR
IS TIME BIAS ERROR CONSIDERED
TIME BIAS VARIANCES SET ZERO
COMPUTE RANGE, AZIMUTH & ELEVATION VARIANCES DUE TO TIME BIAS
WRITE OUTPUT STATION DATA AZ - EL
WRITE OUTPUT STATION DATA DIRCOS
WHAT TYPE OF ANGLE DATA IS BEING MEASURED
AZ-EL NONE
DIR-COS
IS PRINTOUT DESIRED
IS RANGE MEASUREMENT CONSIDERED
IS AZIMUTH ELEVATION MEASUREMENT CONSIDERED
IS DIRCOS MEASUREMENT CONSIDERED
IS RANGE RATE MEASUREMENT CONSIDERED
COMPUTE H MATRIX PARTIALS FOR RANGE
COMPUTE RANGE VARIANCE DUE TO STATION LOCATION ERRORS
COMPUTE TOTAL RANGE VARIANCE
CALL TRANSH TRANSFORM PARTIALS FROM DATE TO 1950
CALL CORRTP UPDATE COV. MATRIX FOR MEASUREMENT
COMPUTE H MATRIX PARTIALS FOR AZ
COMPUTE AZ VARIANCE DUE TO STATION LOCATION ERRORS
COMPUTE TOTAL AZ VARIANCE
CALL TRANSH
WHAT TYPE OF MEASUREMENT IS BEING MADE AZ EL DIRCOS
CALL CORRTP
COMPUTE H MATRIX PARTIALS FOR EL
COMPUTE EL VARIANCE DUE TO STATION LOCATION ERRORS
COMPUTE TOTAL EL VARIANCE
CALL TRANSH
WHAT TYPE OF MEASUREMENT IS BEING MADE AZ EL DIRCOS
CALL CORRTP
COMPUTE TRANSFORMATION FROM AZ EL TO M DIRCOS
COMPUTE H MATRIX PARTIALS FOR M DIRCOS
COMPUTE TOTAL M DIRCOS VARIANCE
CALL CORRTP
COMPUTE TRANSFORMATION FROM AZ EL TO L DIRCOS
COMPUTE H MATRIX PARTIALS FOR L DIRCOS
COMPUTE TOTAL L DIRCOS VARIANCE
CALL CORRTP
COMPUTE H MATRIX PARTIALS FOR RR
COMPUTE RR VARIANCE DUE TO STATION LOCATION ERRORS
COMPUTE TOTAL RR VARIANCE
CALL TRANSH
CALL CORRTP

EARTR

```
*       LABEL                                                          EART
*       SYMBOL TABLE                                                   EART
        SUBROUTINE EARTR                                               EART0000
C       SUBROUTINE EARTR UPDATES THE COVARIANCE MATRIX USED            EART0005
C       IN EARTH TRACKING                                              EART0006
        COMMON T,S,C,IC                                                EART0010
        DIMENSION  PO(22),VE(22),XED(3),VED(3),AN(3,3),RBOP(6),POSM(3), EART0020
       1VOSM(3),POVRM(3),EM(3,3),EN(3),U(3), E(3),RT(3),Y(3),TD(3),     EART0030
       2ISTAT(20),H(6),X(3),VX(3),T(1360),S(1000),C(1000),IC(1),ISA(20), EART0040
       3IMSTA(4,20),P(6,6),SPART(3),DRT(3,3),XD(3),SPART2(3),HA(6),HCOS(6)EART0050
        EQUIVALENCE (T,TDUM),(S,SDUM),(C,CDUM)                         EART0060
       1,(IC(3),NOR),(C(13),TW),(C(14),TF),(C(62),PO),(C(84),VE)       EART0070
       2,(C(112),RBOP),(C(138),AN),(C(120),EM),(C(15),X),(C(18),VX),   EART0080
       3(S(70),DTS),(S(71),STD),(S(72),DR),(S(73),RD),(IC(194),ISTAT)  EART0090
       4,(IC(10),ISA),(IC(30),IMSTA),(IC(214),NOUT),(C(652),P)         EART0100
       5,(C(649),TSECP),(C(30),TSEC),(S(10),W),(S(14),A),(S(15),B)     EART0110
        CALL SETN(NIN,NUTS)                                            EART0115
        NB=NOR-1                                                       EART0120
        DIS=1.E10                                                      EART0130
        CALL INTR1 (TW,TF,NB,PO,1,VE,DIS)                              EART0140
        CALL MATRX(AN,X,XED,3,3,1,0)                                   EART0150
        CALL MATRX(AN,VX,VED,3,3,1,0)                                  EART0160
C       XED=VECTOR FROM NB BODY TO VEHICLE, VED=VELOCITY EQUATOR OF DATE EART0170
        MS=1                                                           EART0180
        IF(RBOP(1)-330000.)10,2,2                                      EART0190
C       RBOP= MAGNITUDE OF VECTOR FROM VEHICLE TO BODY                 EART0200
    2   CONTINUE                                                       EART0210
        IF (NOR-1)3,5,3                                                EART0220
    3   CONTINUE                                                       EART0230
C       HERE IF OUTSIDE MOON AND NOT EARTH REFERENCED                  EART0240
        DO 4 I=1,3                                                     EART0250
        POSM(I)=0.                                                     EART0260
        VOSM(I)=0.                                                     EART0270
        DO 4 J=1,3                                                     EART0280
        K=23-J                                                         EART0290
        XED(I)=XED(I)-AN(I,J)*PO(K)                                    EART0300
        VED(I)=VED(I)-AN(I,J)*VE(K)                                    EART0310
        L=20-J                                                         EART0320
        POSM(I)=POSM(I)+AN(I,J)*(PO(L)-PO(K))                          EART0330
        VOSM(I)=VOSM(I)+AN(I,J)*(VE(L)-VE(K))                          EART0340
C       POSM IS A VECTOR FROM EARTH TO MOON                            EART0350
    4   CONTINUE                                                       EART0360
        GO TO 7                                                        EART0370
    5   CONTINUE                                                       EART0380
C       HERE IF OUTSIDE MOON AND EARTH REFERENCED                      EART0390
        DO 6 I=1,3                                                     EART0400
        POSM(I)=0                                                      EART0410
        VOSM(I)=0                                                      EART0420
        DO 6 J=1,3                                                     EART0430
        K=20-J                                                         EART0440
        POSM(I)=POSM(I)+AN(I,J)*PO(K)                                  EART0450
```

EARTR (Cont'd)

```
      VOSM(I)=VOSM(I)+AN(I,J)*VE(K)
    6 CONTINUE
    7 CONTINUE
      MS=2
      DO 8 I=1,3
      POVRM(I)=XED(I)-POSM(I)
    8 CONTINUE
      RADM=FNORM(POVRM)
      DO 9 I=1,3
      POVRM(I)=POVRM(I)/RADM
    9 CONTINUE
   10 CONTINUE
      TIME=TW+TF
      CALL       NUTAIT (TIME,OM,CR,DT,EM,ERSIL)
      FSEC=DTS*TF
      CALL GHA(FSEC,TW,GH,EM(2,1),OMEGA)
      GH=GH*DR
      NDUM=783
      NN=110
      DO 53 III=1,20
      NDUM=NDUM+5
      NN=NN+15
      IF(ISA(III)) 53,53,11
   11 CONTINUE
      AT=S(NN+9)*DR
      ON=S(NN+10)*DR
      AL=S(NN+11)
      SNAME=S(NN+12)
      CALL TRAC(ON,AT,AL,GH,U,E,EN,RT,AC,SL,CL,ST,CT,A,B)
C     TRAC  OBTAINS UNIT VECTORS UP,EAST AND NORTH, RT= VECTOR
C     TO TRACK STATION
      DO 12 J=1,3
      Y(J)=XED(J)-RT(J)
C     Y=VECTOR FROM TRACK STATION TO VEHICLE. TD=UNIT VECTOR IN
C     DIRECTION OF Y
   12 CONTINUE
      RAT=FNORM(Y)
      DO 13 J=1,3
      TD(J)=Y(J)/RAT
   13 CONTINUE
      CE=DOT( U,TD)
C     DOT FORMS A DOT PRODUCT OF ELEMENTS IN PARENTHESES
      SEC=CE/SQRTF(1.-CE**2)
      ELEV=ATANF(SEC)
      HORCOR=S(NDUM)
C     S(NDUM) CONTAINS HORIZON CORRECTION FOR STATION
      IF(ELEV-HORCOR)22,14,14
   14 CONTINUE
C     HERE IF VEHICLE IS ABOVE STATION HORIZON
      GO TO (20,15),MS
   15 CONTINUE
```

EARTR (Cont'd)

```
C      TO DETERMINE IF MOON OCCULTS VEHICLE                                    EART0930
       DOTTD=DOT(POVRM,TD)                                                     EART0940
       IF(DOTTD)19,19,16                                                       EART0950
   16  CONTINUE                                                                EART0960
       IF(DOTTD-(SQRTF(RADM**2-1739.**2))/RADM)19,19,17                        EART0970
   17  WRITE OUTPUT TAPE NUTS,18,SNAME                                         EART0980
   18  FORMAT (27H0MOON OCCULTS VEHICLE FROM,A6,8H TRACKER)                    EART0990
       GO TO 22                                                                EART1000
   19  CONTINUE                                                                EART1010
   20  CONTINUE                                                                EART1020
       SEC=DOT(TD,EN)                                                          EART1030
       CE=DOT(TD,E)                                                            EART1040
       AZMTH =ARKTNS(360,SEC,CE)                                               EART1050
       IF(ISTAT(III))21,21,23                                                  EART1060
C ISTAT IS SET ONE IF THE STATION SAW VEHICLE ON PREVIOUS TIME                 EART106[illegible]
C      THROUGH MATSUB                                                          EART1066
C      ZERO IF IT COULD NOT SEE IT                                             EART1070
   21  CONTINUE                                                                EART1080
C      HERE IF STATION SEES THE VEHICLE                                        EART1090
       ISTAT(III)=1                                                            EART1100
       GO TO 53                                                                EART1110
   22  CONTINUE                                                                EART1120
       ISTAT(III)=0                                                            EART1130
C      VEHICLE WAS NOT OBSERVED BY STATION                                     EART1140
       GO TO 53                                                                EART1150
   23  CONTINUE                                                                EART1160
       OBNO=INTF((TSEC-TSECP)/S(NN+13))                                        EART1170
C      OBNO IS THE NUMBER OF OBSERVATIONS SINCE LAST TIME IN MATSUB            EART1180
       XTU=DOT(Y, U)                                                           EART1190
       XTE=DOT(Y,E)                                                            EART1200
       XTN=DOT(Y,EN)                                                           EART1210
       TPD2=XTE**2+XTN**2                                                      EART1220
       TPD=SQRTF(TPD2)                                                         EART1230
       ALLY=XTN/TPD2                                                           EART1240
       GAM=XTU*XTE                                                             EART1250
       BET=XTE/TPD2                                                            EART1260
       DEL=XTU*XTN                                                             EART1270
       D=1./RAT**2/TPD                                                         EART1280
       CA=AC/A                                                                 EART1290
       BA=B*B/(A*A)                                                            EART1300
       GG=CA*(1.-BA)*SL*CL*CA*CA                                               EART1310
       DRT(1,1)=A*GG*U(1)+(AC+AL)*EN(1)                                        EART1320
       DRT(2,1)=A*GG*U(2)+(AC+AL)*EN(2)                                        EART1330
       DRT(3,1)=(B*B/A)*GG*U(3)+BA*(AC+AL)*EN(3)                               EART1340
       DRT(1,2)=(AC+AL)*CL*E(1)                                                EART1350
       DRT(2,2)=(AC+AL)*CL*E(2)                                                EART1360
       DRT(3,2)=0.0                                                            EART1370
       DO 24 I=1,3                                                             EART1380
       DRT(I,3)=U(I)                                                           EART1390
C      THE ABOVE    QUATITIES ARE USED ONLY FOR H MATRIX CALCULATIONS          EART1400
   24  CONTINUE                                                                EART1410
```

EARTR (Cont'd)

```
      AA=S(10)*CL*(AC+AL)                                               EART142
      XD(1)=VED(1)-AA*E(1)                                              EART143
      XD(2)=VED(2)-AA*E(2)                                              EART144
      XD(3)=VED(3)                                                      EART145
      XDTU=DOT(XD,U)                                                    EART146
      XDTE=DOT(XD,E)                                                    EART147
      XDTN=DOT(XD,EN)                                                   EART148
      RDOT=DOT(XD,Y)/RAT                                                EART149
C     RDOT=RANGE RATE                                                   EART150
      ADOT=W*SL-W*CL*ALLY*XTU+ALLY*XDTE-BET*XDTN                        EART151
C     ADOT=AZIMUTH RATE                                                 EART152
      EDOT=(1./TPD)*(W*CL*XTE+XDTU-RDOT*XTU/RAT)                        EART153
C     EDOT=ELEVATION RATE                                               EART154
      IF(S(NN+14))26,26,25                                              EART155
   25 CONTINUE                                                          EART156
      RDOT2=RDOT*RDOT                                                   EART157
      QQRDOT=RDOT2*S(NN+14)                                             EART158
C     QQRDOT=RANGE VARIANCE DUE TO TIME BIAS                            EART159
      ADOT2=ADOT*ADOT                                                   EART160
      QQADOT=ADOT2*S(NN+14)                                             EART161
C     QQADOT=AZIMUTH VARIANCE DUE TO TIME BIAS                          EART162
      EDOT2=EDOT*EDOT                                                   EART163
      QQEDOT=EDOT2*S(NN+14)                                             EART164
C     QQEDOT=ELEVATION VARIANCE DUE TO TIME BIAS                        EART165
      GO TO 27                                                          EART166
   26 CONTINUE                                                          EART167
      QQRDOT=0.                                                         EART168
      QQADOT=0.                                                         EART169
      QQEDOT=0.                                                         EART170
   27 CONTINUE                                                          EART171
      DO 49 II=1,4                                                      EART172
      IF (IMSTA(II,III)) 49,49,28                                       EART173
   28 CONTINUE                                                          EART174
      II=II                                                             EART175
      GO TO (29,35,35,45),II                                            EART176
C     II DESIGNATES TYPE OF MEASUREMENT 1=RANGE 2=AZ+EL 3=DIRCOS        EART176
C     4=RANGE RATE                                                      EART176
   29 CONTINUE                                                          EART177
C     CALCULATES H MATRIX FOR RANGE                                     EART178
      DO 30 J=1,3                                                       EART179
      H(J)=TD(J)                                                        EART180
   30 CONTINUE                                                          EART181
      DO 32 I=1,3                                                       EART182
      SPART(I)=0.                                                       EART183
C     SPART=PARTIALS OF RANGE WRT STATION LOCATION                      EART184
      DO 31 K=1,3                                                       EART185
      SPART(I)=SPART(I)-H(K)*DRT(K,I)                                   EART186
   31 CONTINUE                                                          EART187
      SPART2(I)=SPART(I)*SPART(I)                                       EART188
   32 CONTINUE                                                          EART189
      QQS=0.                                                            EART190
```

EARTR (Cont'd)

```
      DO 33 I=1,3                                                       EART1910
C     QQS=RANGE VARIANCE DUE TO STATION LOCATION ERRORS                 EART1920
      KL=NN+3+I                                                         EART1930
      QQS=QQS+SPART2(I)*S(KL)                                           EART1940
   33 CONTINUE                                                          EART1950
      QQ=S(NN)/OBNO+RAT*S(425)*RAT+QQS+QQRDOT                           EART1960
C     QQ=TOTAL RANGE VARIANCE                                           EART1970
      CALL TRANSH(H,1)                                                  EART1980
C     TRANSH TRANSFORMS H MATRIX FROM EQUINOX OF DATE TO 1950           EART1990
   34 CONTINUE                                                          EART2000
      CALL CORRTP (QQ,H,P)                                              EART2010
C     CORRTP UPDATES COVARIANCE MATRIX FOR PARTICULAR TYPE              EART2015
C     OF MEASUREMENT                                                    EART2016
      GO TO 49                                                          EART2020
   35 CONTINUE                                                          EART2030
C     THE FOLLOWING CALCULATIONS GIVE H MATRIX FOR AZIMUTH              EART2040
      DO 36 J=1,3                                                       EART2050
      HA(J)=ALLY*E(J)-BET*EN(J)                                         EART2060
   36 CONTINUE                                                          EART2070
      DO 37 I=1,3                                                       EART2080
      SPART(I)=0.                                                       EART2090
C     SPART=PARTIALS OF AZIMUTH WRT STATION LOCATION                    EART2100
      DO 37 K=1,3                                                       EART2110
      SPART(I)=SPART(I)-HA(K)*DRT(K,I)                                  EART2120
   37 CONTINUE                                                          EART2130
      SPART(1)=SPART(1)+BET*XTU                                         EART2140
      PLUS=BET*SL*XTE-ALLY*(Y(1)*CT+Y(2)*ST)                            EART2150
      SPART(2)=SPART(2)+PLUS                                            EART2160
      DO 38 I=1,3                                                       EART2170
      SPART2(I)=SPART(I)*SPART(I)                                       EART2180
   38 CONTINUE                                                          EART2190
      QQSA=0.                                                           EART2200
      DO 39 I=1,3                                                       EART2210
C     QQSA=AZIMUTH VARIANCE DUE TO STATION LOCATION ERRORS              EART2220
      KL=NN+3+I                                                         EART2230
      QQSA=QQSA+SPART2(I)*S(KL)                                         EART2240
   39 CONTINUE                                                          EART2250
      QQ=S(NN+1)/OBNO+S(NN+7)+QQSA+QQADOT                               EART2260
C     QQ=TOTAL AZIMUTH VARIANCE                                         EART2270
      CALL TRANSH(HA,1)                                                 EART2280
      IF(II-2)54,54,40                                                  EART2290
   54 CONTINUE                                                          EART2300
      CALL CORRTP(QQ,HA,P)                                              EART2310
   40 CONTINUE                                                          EART2320
C     THE FOLLOWING CALCULATIONS GIVE H MATRIX FOR ELEVATION            EART2330
      DO 41 J=1,3                                                       EART2340
      H(J)= U(J)/TPD-Y(J)*XTU*D                                         EART2350
   41 CONTINUE                                                          EART2360
      DO 42 I=1,3                                                       EART2370
      SPART(I)=0.                                                       EART2380
C     SPART=PARTIALS OF ELEVATION WRT STATION LOCATION                  EART2390
```

EARTR (Cont'd)

```
      DO 42 K=1,3
      SPART(I)=SPART(I)-H(K)*DRT(K,I)
   42 CONTINUE
      SPART(1)=SPART(1)+XTN/TPD
      SPART(2)=SPART(2)+XTE*CL/TPD
      DO 43 I=1,3
      SPART2(I)=SPART(I)*SPART(I)
   43 CONTINUE
      QQS=0.
      DO 44 I=1,3
C     QQS=ELEVATION VARIANCE DUE TO STATION LOCATION ERRORS
      KL=NN+3+I
      QQS=QQS+SPART2(I)*S(KL)
   44 CONTINUE
      QQ=S(NN+2)/OBNO+S(NN+8)+QQS+QQEDOT
C     QQ=TOTAL ELEVATION VARIANCE
      CALL TRANSH(H,1)
      IF(II-2)34,34,55
   55 CONTINUE
      DIRCOS=DOT(EN,TD)
      SINANG=SQRTF(1.-DIRCOS**2)
      TANANG=SINANG/DIRCOS
      ANGM=ATANF(TANANG)
      ANGM=ANGM*RD
      CA=COSF(AZMTH)
      SA=SINF(AZMTH)
      SE=SINF(ELEV)
      CE=COSF(ELEV)
      C1=-SA*CE
      C2=-CA*SE
      ANGMDT=C1*EDOT+C2*ADOT
      ANGMDT=ANGMDT*RD
      DO 56 I=1,3
      HCOS(I)=C1*H(I)+C2*HA(I)
   56 CONTINUE
      QQSCOS=C1*QQS+C2*QQSA
C     QQSCOS IS THE VARIANCE IN THE M DIRECTION COSINE DUE TO STATION
C     LOCATION ERRORS
      QQCDOT=C1*QQEDOT+C2*QQADOT
      QQ=S(NDUM+1)/OBNO+S(NDUM+2)+QQSCOS+QQCDOT
C     QQ=TOTAL VARIANCE OF M DIRECTION COSINE
      CALL CORRTP(QQ,HCOS,P)
   57 CONTINUE
      DIRCOS=DOT(E,TD)
      SINANG=SQRTF(1.-DIRCOS**2)
      TANANG=SINANG/DIRCOS
      ANGL=ATANF(TANANG)
      ANGL=ANGL*RD
      C1=CA*CE
      C2=-SA*SE
      ANGLDT=C1*EDOT+C2*ADOT
```

EARTR (Cont'd)

```
      ANGLDT=ANGLDT*RD
      DO 58 I=1,3
      HCOS(I)=C1*H(I)+C2*HA(I)
   58 CONTINUE
      QQSCOS=C1*QQS+C2*QQSA
C     QQSCOS IS THE L DIRECTION COSINE ERROR DUE TO STATION LOCATION
C     ERROR
      QQCDOT=C1*QQEDOT+C2*QQADOT
      QQ=S(NDUM+3)/OBNO+S(NDUM+4)+QQSCOS+QQCDOT
C     QQ=TOTAL L DIRECTION COSINE VARIANCE
      CALL CORRTP(QQ,HCOS,P)
      GO TO 49
   45 CONTINUE
C     THE FOLLOWING CALCULATIONS GIVE H MATRIX FOR RANGE RATE
   47 DO 48 J=1,3
      K=J+3
      H(K)=TD(J)
      H(J)=(1./RAT)*(XD(J)-RDOT*TD(J))
   48 CONTINUE
      CNST=B*B/A*((1./CA)**3)
      DO 65 J=1,3
      JJ=3*J-2
      SPART(J)=-DOT(H,DRT(JJ))
   65 CONTINUE
      SPART(1)=SPART(1)+S(10)*U(3)*(CNST+AL)*XTE/RAT
      SPART(2)=SPART(2)-AA*(TD(2)*E(1)-TD(1)*E(2))
      SPART(3)=SPART(3)-S(10)*EN(3)*XTE/RAT
      QQS=0.
      DO 66 I=1,3
      KL=NN+3+I
      QQS=QQS+(SPART(I)**2)*S(KL)
   66 CONTINUE
      QQ=S(NN+3)/OBNO+QQS
      CALL TRANSH(H,2)
      GO TO 34
   49 CONTINUE
      NOUT=NOUT
      GO TO (52,50),NOUT
   50 CONTINUE
      IF(IMSTA(3,III))59,59,60
   59 CONTINUE
      EDOT=EDOT*RD
      ADOT=ADOT*RD
      AZMTH =AZMTH*RD
      ELEV=ELEV*RD
      WRITE OUTPUT TAPE NUTS,51,SNAME,RAT,RDOT,AZMTH,ADOT,ELEV,EDOT
   51 FORMAT (17H TRACKER STATION ,A6,/4H RNGE15.8,5H  RGRE15.8,
     15H  AZME15.8,5H  AZRE15.8,5H  ELEE15.8,5H  ELRE15.8)
      GO TO 52
   60 CONTINUE
      WRITE OUTPUT TAPE NUTS,61,SNAME,RAT,RDOT,ANGL,ANGLDT,ANGM,ANGMDT
```

EARTR (Cont'd)

```
 61  FORMAT (17H TRACKER STATION ,A6,/4H RNGE15.8,5H  RGRE15.8,       EART32
    15H  LCSE15.8,5H  LCRE15.8,5H  MCSE15.8,5H  MCRE15.8)             EART33
  52 CONTINUE                                                          EART33
     GO TO 21                                                          EART33
  53 CONTINUE                                                          EART33
     RETURN                                                            EART33
     END                                                               EART
```

Page Intentionally Left Blank

Subroutine: ECLIP

Purpose: To replace each of two 3-dimensional vectors by its product with a given 3 x 3 matrix. This subroutine is used for transforming position and velocity vectors.

Calling Sequence:

CALL ECLIP (X, VX, ECL)

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I-O	X	3			Xout = (ECL) Xin
I-O	VX	3			VXout = (ECL) VXin
I	ECL	3,3			TRANSFORMATION MATRIX

Common storages used or required: None

Subroutines required: None

Functions required: None

Approximate number of storages required:

```
*     LABEL                                                      ECLP
*     SYMBOL TABLE                                               ECLP
      SUBROUTINE ECLIP(X,VX,ECL)                                 ECLP0000
      DIMENSION X(3),VX(3),XP(3),VXP(3),ECL(3,3)                 ECLP0010
      DO 1 I=1,3                                                 ECLP0020
      XP(I) = 0.                                                 ECLP0030
      VXP(I) = 0.                                                ECLP0040
      DO 1 J=1,3                                                 ECLP0050
      XP(I) = XP(I) + ECL(I,J)*X(J)                              ECLP0060
    1 VXP(I) = VXP(I) + ECL(I,J)*VX(J)                           ECLP0070
      DO 2 I=1,3                                                 ECLP0080
      X(I) = XP(I)                                               ECLP0090
    2 VX(I) = VXP(I)                                             ECLP0100
      RETURN                                                     ECLP0110
      END                                                        ECLP
```

Subroutine: ENCKE

Purpose: To perform the calculation of the ENCKE contribution to the acceleration. These are perturbations in acceleration due to deviation from the osculating conic.

Calling Sequence:

CALL ENCKE (U, X, D, AE)

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	U	1	μ	Km^3/sec^2	Gravitational constant
I	X	3	X,Y,Z	Km	Position on the conic
I	D	3	X',Y',Z'	Km	Position deviation from the conic
O	AE	3		Km/sec^2	ENCKE acceleration terms

Common storages used or required: None

Subroutines required: None

Functions required: SQRT, DOT

Approximate number of storages required: 158 DEC

Equation Being Solved

The perturbation acceleration due to deviation from the reference conic is the following

$$\vec{AE} = \frac{\mu}{R_o^3} (\vec{R}\ F(Q) - \vec{D})$$

where $\vec{R} = \vec{R}_o + \vec{D}$. The solution, R_o, for the position in the two body orbit is provided by STEPC, and is saved from step to step so that a new R_o is calculated only when the time has changed.

$F(Q) = 1 - (1 + 2Q)^{-3/2}$ is calculated from the series expansion

$$F(Q) = Q \sum_{j=0}^{6} a_j Q^j$$

where

$a_0 = 3$

$a_1 = -7.5$

$a_2 = 17.5$

$a_3 = 39.375$

$a_4 = 86.625$

$a_5 = 187.6875$

$a_6 = 402.1875$

and

$$Q = \frac{\vec{D} \cdot \left(\vec{R_o} + \frac{\vec{D}}{2}\right)}{R_o^2}$$

ENCKE

```
*       LABEL                                                        ENCK
*       SYMBOL TABLE                                                 ENCK
CEC2014  SUBROUTINE ENCKE                                            ENCK
        SUBROUTINE ENCKE (U,X,D,AE)                                  ENCK00
        DIMENSION X(3),D(3),AE(3)                                    ENCK00
        R = DOT(X,X)                                                 ENCK00
        DD = DOT(D,D)                                                ENCK00
        RD = DOT(X,D)                                                ENCK00
        Q = (RD+.5*DD)/R                                             ENCK00
        F = 1./SQRTF(2.*Q+1.)**3                                     ENCK01
      2 A = Q*(3.+Q*(-7.5+Q*(17.5+Q*(-39.375+Q*(86.625+Q*(-187.6875+Q*  ENCK01
       1402.1875))))))                                               ENCK01
      4 R3 = (SQRTF(R))*R                                            ENCK01
        UA = U*A/R3                                                  ENCK01
        UF = U*F/R3                                                  ENCK01
        DO 5 I=1,3                                                   ENCK01
      5 AE(I) = UA*X(I)-UF*D(I)                                      ENCK01
        RETURN                                                       ENCK01
        END                                                          ENCK
```

Page Intentionally Left Blank

Subroutine: ERP

Purpose: To write on the output tape: EPHEMERIS FAILED DUE TO TAPE REDUNDANCIES, and then to CALL EXIT. Should this message occur, and the tape was actually mounted on the correct unit, try again or try a different copy of the ephemeris tape.

Calling Sequence:

CALL ERP

Input and **Output**

Common storages used or required: None

Subroutines required: SETN, EXIT

Functions required:

Approximate number of storages required:

```
*       LABEL                                                         ERPO
CEC20GD                                                               ERPO
        SUBROUTINE ERP                                                ERP0000
        CALL SETN(NIN,NOUT)                                           ERP0010
  1     FORMAT(42H EPHEMERIS FAILED DUE TO TAPE REDUNDANCIES)         ERP0020
        WRITE OUTPUT TAPE NOUT,1                                      ERP0030
        CALL EXIT                                                     ERP0040
        RETURN                                                        ERP0050
        END                                                           ERPO
```

Subroutine: ERPT

Purpose: To write on the output tape: EPHEMERIS FAILED LOOKING FOR T = (   ) DAYS SINCE 1950.0, and to call EXIT.

The cause of this condition is either incorrect input or not having the ephemeris tape mounted correctly.

Calling Sequence:

CALL ERPT(T)

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	T			days	Time (in days since 1950.0)
					for which INTR was searching
					the tape

Common storages used or required: None

Subroutines required: SETN, EXIT

Functions required: None

Approximate number of storages required:

```
*     LABEL                                                               ERPT
CEC20GE                                                                   ERPT
      SUBROUTINE ERPT(T)                                                  ERPT0000
      CALL SETN(NIN,NOUT)                                                 ERPT0010
      WRITE OUTPUT TAPE NOUT,2,T                                          ERPT0020
 2    FORMAT(33H EPHEMERIS FAILED LOOKING FOR T =,F20.0,                  ERPT0030
     118H DAYS SINCE 1950,0)                                              ERPT0040
      CALL EXIT                                                           ERPT0050
      RETURN                                                              ERPT0060
      END                                                                 ERPT
```

Subroutine: FINP

Identification

RW FINP - Decimal, Octal, BCD, Variable Data Input
7090 FAP Subroutine
W.J. Stoner, August 24, 1961
Aerospace Corporation

Purpose

To read a set of Hollerith punched data and/or header cards into core with one FORTRAN CALL statement.

To convert the data fields to binary and store in core according to their associated conversion codes.

Restrictions

This routine uses (CSH)S to accomplish the BCD card image read. Tape troubles or other errors from this routine are indicated by the printout of HPR 1,4.

This routine uses (EXE) to print HPR 2,4 in case of errors such as non-Hollerith characters, data out of range, illegal format, subscripts too large for the array previously defined, etc. Upon detection of any error, control is immediately sent to (EXE) and no more cards are processed.

Method

Decimal numbers are converted to binary integers and then scaled to the indicated power of ten.

Octal numbers are converted to binary integers.

Hollerith words are stored directly.

Range: Decimal to floating binary conversion $10^{\pm 38}$
Decimal to fixed binary; 1 to 9 digits*
Decimal integer to binary integer; 1 to 5 digits
Octal integer to binary integer; 0 to 235 - 1

*the magnitude of the number depends upon the location of the decimal point.

Usage

Format:

1. The data card format, available on keypunch form M-1, consists of four subfields containing the conversion code, location, number, and exponent, respectively.

	Data Field	Data Field	Data Field	Data Field
Subfield	1	2	3	4
Conversion Code	1	19	37	55
Location	2-6	20-24	38-42	56-60
Value	7-16	25-34	43-52	61-70
Exponent	17-18	35-36	53-54	71-72

where conversion code is one of the alphabetic characters defined below which specifies the type of conversion to be used on the value field, the location specifies the cell into which the converted value field is to be stored, the value subfield contains the data to be converted, and the exponent contains the power of ten by which floating data is to be scaled, or the location of the binary point of fixed point data.

2. The header card format consists of a conversion code in column 1, a sequence number in columns 2-6, and any Hollerith information in columns 7-72.

Decimal Points:

Decimal points may be placed anywhere in the value field except that they may not occur in the same column as a minus sign (11 punch) since this results in a non-Hollerith character. If the decimal point would normally appear at the right of the number punched in the value field, then it is optional.

Minus Signs:

Minus signs are 11 punches over any digit of the field. If all of the available columns of the field are not used, minus signs may be punched as the left character of the field.

Values:

Values must always be written to the extreme left of a field. It is not necessary that the entire field be filled as the first blank denotes the end of value. Superfluous low order zeros should be omitted as they increase conversion error.

The only exception to partial fields is BCK data where the entire field, including blanks, is stored.

Location:

The location may be specified by either absolute octal, a variable or array name, or the element subscripts in a one or two dimensional array. If the locations contain five digits, it is interpreted as octal. All five columns must be punched for octal locations.

If the location contains at least one non-numeric character, it is interpreted as a variable or array name which must appear exactly as given in the CALL statement (see Calling Sequence below). The contents of the number and exponent fields, if they are numeric data, are stored in the cell for the variable or the first cell for the array. This location then becomes the origin for all subscript locations following until another variable or array name is encountered. Caution must be taken to load an array name prior to subscript locations.

If the location contains four or fewer digits, it is interpreted as a subscript except for conversion code H explained below. Single dimension array subscripts must be left-justified with leading zeros optional. Two dimension array subscripts must be denoted by two subfields of two columns each containing i and j, respectively. The i and j subfields must be separated by a comma and must contain two-digit integers.

If the location is left blank, then the location counter within the routine is decreased by 1 and the associated number is stored in the cell immediately preceding the cell where the last number was stored. Thus, an entire array may be read in by specifying the initial location only.

Conversion Codes:

Blank: Floating decimal

The number in the value field times the power of ten in the exponent field is converted to floating binary. Checks are made for overflow and format errors.

F: Fixed decimal

The number in the value field is converted to fixed point binary and stored with the binary point located at the position specified by the number in the exponent field. An overflow error check is made.

I: Decimal integer

The number in the value field is converted to a fixed point binary integer with the binary point following position 17. The exponent field is ignored. A decimal point is considered an error.

B: Octal

The value plus exponent fields are converted as a logical ocatal word.

It is not necessary to include leading zeros but the first octal digit must always occupy the leftmost position of the field.

D: BCI Data

The contents of the value plus exponent fields are interpreted as two BCI words and stored in two consecutive cells in descending order beginning at the location specified by the location field.

H: Heading Card

A card with an H in column 1 is considered a BCI heading card. If the location field is blank, the card is ignored. If the location field contains a left-justified one to four digit positive decimal integer V (octal, negative, or variable locations are not permitted), columns 7-72 of the card are stored directly in 11 consecutive words in descending order. The location of the first of these words is calculated by the routine as HEAD (1+11*(V-1)) where HEAD is defined as the last variable or array named in the CALL statement. Each card may be used as one record of output using FORMAT option A with column 7 of the card providing the code for printer spacing on output.

A: Variable names as data

The value plus exponent fields are interpreted in a pseudo FAP instruction format AAAAA T DDDDD P where the fields to replace are address, tag, decrement, and prefix, respectively. The address and decrement fields are defined normally to be 5 characters and the tag and prefix as one octal numeric character each. Any field containing less than the normal number of characters must end with a comma while fields of normal length must not. Any address or decrement field containing less than 5 numeric characters are converted as octal. Any address or decrement field containing at least one non-numeric character is interpreted as a variable or array name. Variable addresses cause the entire word from the compiler generated calling sequence to be loaded into the location word (i.e., the TSX X is stored in the location specified if X is the variable appearing in the address field). Variable decrements cause the right-most 18 bits from the compiler generated calling sequence to be loaded into the location word's prefix and decrement. Numeric tags and prefixes are loaded directly into the corresponding parts of the location words. Null fields are not loaded. Since the first blank indicates the end of the loading of a word, address only, adress-tag, address-tag-decrement, or entire word may be loaded as desired.

G: Temporary Origin

The value in the first location field on the card is used as a temporary origin for tables. The location is saved and if data cards follow with blank location fields the corresponding data is stored consecutively in descending order beginning with the cell specified in the location in the G card. Columns 7-72 are ignored and may be used to identify the table.

The first nonblank location starts a new origin. If this nonblank location is a subscript, it references the last variable or array named, which may or may not have been on the G card.

J: Transfer

The location specified with this prefix must be an octal address and is the only part of the data field that is interpreted. The subroutine causes a transfer to the octal location specified and does not interpret the remaining fields on the card.

L: Two dimension array i_{max}, j_{max} definition

The location field contains the name of the array to be loaded. The value field is defined to consist of 2 subfields, separated by a comma, of 2 columns each containing the two-digit decimal integers for i_{max} and j_{max} respectively where i_{max} and j_{max} generally appear in a DIMENSIØN statement. The i_{max} and j_{max} values are retained to compute the successive subscripted locations until redefined. Blank address fields may follow this array definition if successive elements of the array are to be loaded.

M: Two dimension array i_{max}, j_{max} definition

Conversion is identical to L except the entire array is preset to zero.

E: End Case

This defines an end-of-case and control is returned to the FORTRAN object program. The rest of this field and the remaining fields on the card are ignored.

Calling Sequence:

The following two types of CALL statements may be used:

I. CALL FINP (n,X,Y,ZETA,...,mHX 5 Y 5 ZETA 2 ...) where

   A. n is the number of variables and/or arrays in the list, excluding n itself.

   B. X, Y, ZETA,... are the names of variables and/or arrays restricted to at most 5 characters each, one character of which is non-numeric.

   C. m is 6 times n. Hence, mH allows for 6n Hollerith characters to follow.

D. X(5)Y(5)ZETA(2)... is a list of the items previously named in exactly the same order with (i) indicating the number, i, of blanks necessary to provide six Hollerith characters for each item. Since each item name is restricted to 5 characters, the minimum value of (i) is (1).

II. CALL FINP (0) where the number of items is given as zero. This CALL statement must be used only after a CALL statement of type I has been executed. When the subroutine encounters a zero for the number of items, it immediately refers to the last executed CALL FINP with a nonzero number of items for the names of the items to be loaded.

Space Requirements

613 cells.

Number of Pages

Writeup	6
Listing	12
Total	18

FINP

```
*        FAP                                                             FINP
         LBL FINP                                                        FINP
         TTL      FINP INPUT ROUTINE                                     FINP
         ENTRY    FINP                                                   FINP01
         REM      CALL FINP (N,A,B,DATA,...,6NHA      B      DATA   ...) FINP000
         REM           WHERE N IS NUM OF VARIABLES AND                   FINP000
         REM           6N IS 6 TIMES NUM OF VARIABLES                    FINP000
FINP     TXH      0,0,0                                                  FINP000
         SXD      GIN00,4                                                FINP000
         SXD      GIN02,1                                                FINP000
         SXD      GIN05,2                                                FINP000
         SXD      STOP1,4                                                FINP000
         SXD      STOP2,4                                                FINP001
         CLA*     1,4                                                    FINP001
         TNZ      INPT1                                                  FINP001
         STO      COMMON+28          FLAG TO USE PREVIOUS CALL           FINP001
         TXL      INPT2                                                  FINP001
INPT1    STO      COMMON+28          FLAG TO USE PRESENT CALL            FINP001
         SXD      FINPT,4                                                FINP001
         NOP                                                             FINP001
         COM                                                             FINP001
         STD      *+1                                                    FINP001
         TXI      *+1,4,0                                                FINP002
         SXD      GIN00,4                                                FINP002
INPT2    LXD      FINPT,4                                                FINP002
         CLA*     1,4                                                    FINP002
         ARS      18                                                     FINP002
         STO      COMMON+29          N                                   FINP002
         ALS      18                                                     FINP002
         COM                                                             FINP002
         STD      TBL03                                                  FINP002
         STD      *+1                                                    FINP002
         TXI      *+1,4,0                                                FINP003
         CLA      0,4                                                    FINP003
         STA      GIN01              ADDRESS OF HEADER                   FINP003
         CLA      1,4                                                    FINP003
         STA      TBL01              ADDRESS OF BCD TABLE                FINP003
         STO      ERROR                                                  FINP003
GINA     LXD      GIN00,4                                                FINP003
         CALL     (CSH)S                                                 FINP003
         PZE      COMMON,0,-1                                            FINP003
STOP1    PZE      1,4                                                    FINP003
         AXC      COMMON,1                                               FINP004
         TXI      *+1,1,-12                                              FINP004
         SXD      GINC,1                                                 FINP004
         SXD      GIN11+1,1                                              FINP004
         AXC      COMMON,1                                               FINP004
         CLA      BLANK                                                  FINP004
GINB     CAS      0,1                                                    FINP004
         TXL      GINC+2                                                 FINP004
         TXI      *+2,1,-1                                               FINP004
```

```
        TXL     GINC+2                                                    FINP0049
GINC    TXH     GINB,1,**                                                 FINP0050
        TXL     GINA                                                      FINP0051
        AXC     COMMON,1                                                  FINP0052
        LDQ 0,1                                                           FINP0053
GIN01   PXD **                    SAVE PREFIX CODE                        FINP0054
        LGL 6                                                             FINP0055
        STO PREFIX                                                        FINP0056
        STQ 0,1                                                           FINP0057
        SUB     ME                   E                                    FINP0058
        TZE     GIN13                END                                  FINP0059
        LGL 30                    SAVE LAST DIGIT OF ADDRESS              FINP0060
        ANA MASK1                                                         FINP0061
        STO ADDIND                                                        FINP0062
        STZ SUM                   TEST FOR BLANK ADDRESS                  FINP0063
        STZ SIGN                                                          FINP0064
        CLA     0,1                                                       FINP0065
        CAS     MASK2                                                     FINP0066
        TXL     *+2                                                       FINP0067
        TXL     GIN06                BLANK                                FINP0068
        LRS     18                   TEST FOR MATRIX                      FINP0069
        ANA     MASK1                                                     FINP0070
        SUB     MASK4                                                     FINP0071
        TNZ     GIN17-2                                                   FINP0072
        TSX     MTX00,4              MATRIX                               FINP0073
        TRA     GINER                OUTSIDE MATRIX                       FINP0074
        TRA     GIN20                                                     FINP0075
MASK4   OCT     73                                                        FINP0076
        AXT     5,4                                                       FINP0077
        LDQ     0,1                                                       FINP0078
GIN17   PXD                                                               FINP0079
        LGL     6                                                         FINP0080
        PAX     0,2                                                       FINP0081
        TXL     GIN18,2,9                                                 FINP0082
        TXH     GIN19,2,48                                                FINP0083
        TXL     GIN19,2,47                                                FINP0084
GIN18   TIX     GIN17,4,1                                                 FINP0085
        TXL     GIN16                                                     FINP0086
GIN19   CLA     0,1                 SYMBOLIC                              FINP0087
        ORA     MASK3                                                     FINP0088
        TSX     TBL00,4                                                   FINP0089
        LLS     35                                                        FINP0090
        STA     GIN00                                                     FINP0091
GIN20   CLA     MASK3                                                     FINP0092
        STO     ADDIND                                                    FINP0093
        TXL     GIN07                                                     FINP0094
GIN16   CLA     ADDIND               TEST FOR OCTAL                       FINP0095
        SUB MASK3                                                         FINP0096
        TZE GIN04                                                         FINP0097
        LXA 5B35,2                SET-UP OCTAL ADDRESS                    FINP0098
        TSX OCTAL,4                                                       FINP0099
```

```
GIN02  TXH 5,,**                                                     FINP010
GIN03  TXL GIN07,,**                                                 FINP010
GIN04  LXA 100B35,2           SET UP DECIMAL ADDRESS                 FINP010
       CLA 4B35                                                      FINP010
       TSX DECIM,4                                                   FINP010
GIN05  TXH 6,,**                                                     FINP010
       XCA                                                           FINP010
       SUB     1B35                                                  FINP010
       XCA                                                           FINP010
       TXL GINER,2,100                                               FINP010
       TXL     GIN07                                                 FINP011
GIN06  LDQ MASK2                                                     FINP011
GIN07  STQ CRDADD                                                    FINP011
       LXA PREFIX,4           SET-UP TRA TO                          FINP011
       TXH GINER,4,48            SUB-PROGRAM                         FINP011
       TXL GIN08,4,47                                                FINP011
       LXD 100B35,4                                                  FINP011
       TXL GIN09                                                     FINP011
GIN08  TXH     GINER,4,36                                            FINP011
       TXL     *+2,4,32                                              FINP011
       TXI     GIN09,4,-23                                           FINP012
       TXH     GINER,4,25                                            FINP012
       TNX GINER,4,16                                                FINP012
GIN09  TXI GIN09+1,1,-1                                              FINP012
       TRA GIN10,4                                                   FINP012
       TXL     MCD00             M MATRIX IJ(ZERO                    FINP012
       TXL     MCD00             L MATRIX IJ-NO ZERO                 FINP012
       TXL     GINER             K                                   FINP012
       TRA*    CRDADD                                                FINP012
       TXL     IGIN00            I        INTEGER                    FINP012
       TXL     HGIN00            H        HEADER                     FINP013
       TXL     GGIN00            G        TABLE ORIGIN               FINP013
       TXL     FGIN00            F        FIXED FOINT                FINP013
       TXL GINER              E                                      FINP013
       TXL     DGIN00            D        BCD                        FINP013
       TXL GINER              C                                      FINP013
       TXL     BGIN00            B        OCTAL                      FINP013
       TXL     ACD0              A        SYMBOLIC                   FINP013
GIN10  TXL     LGIN00,,12        BLANK  FLOATINS                     FINP013
GIN11  TXI GIN11+1,1,-1            TEST END CARD                     FINP013
       TXH     GIN01-1,1,**                                          FINP014
GIN12  TXL     GINA              TO NEXT CARD                        FINP014
GIN13  LXD     GIN00,4                                               FINP014
       LXD     GIN02,1                                               FINP014
       LXD     GIN05,2                                               FINP014
       TRA     2,4               EXIT                                FINP014
GINER  PXD                                                           FINP014
       LDI     STOP2                                                 FINP014
       CALL    (EXE)                                                 FINP014
STOP2  PZE     2,4                                                   FINP014
 ME    BCI     1,00000E                                              FINP015
```

```
BLANK   BCI      1,                                          FINP0151
        REM                                                  FINP0152
HGINOO  CLA      CRDADD                                      FINP0153
        CAS      MASK2                                       FINP0154
        TXL      *+2                                         FINP0155
        TXL      GIN12               BLANK                   FINP0156
        LDQ      CRDADD                                      FINP0157
        MPY      11B35                                       FINP0158
        STQ      COMMON                                      FINP0159
        CAL      GINO1                                       FINP0160
        SBM      COMMON                                      FINP0161
        STA HGINO1                                           FINP0162
        LXD      1B35,4                                      FINP0163
        LXA      11B35,1                                     FINP0164
        CLA      COMMON+12,1                                 FINP0165
HGINO1  STO      **,4                                        FINP0166
        TXI      *+1,4,1                                     FINP0167
        TIX HGINO1-1,1,1                                     FINP0168
        TXL GIN12                                            FINP0169
GGINOO  CLA ADDIND                                           FINP0170
        SUB MASK3                                            FINP0171
        TZE GGINO1                                           FINP0172
        PXD                                                  FINP0173
        TXL GGINO2                                           FINP0174
GGINO1  CAL GINOO                                            FINP0175
GGINO2  ADD      1B35                                        FINP0176
        ADD CRDADD                                           FINP0177
        STA SGINO3                                           FINP0178
        TXL GIN12                                            FINP0179
BGINOO  STZ SUM                                              FINP0180
        STZ SIGN                                             FINP0181
        LXA M6B35,2            6                             FINP0182
        TSX OCTAL,4                                          FINP0183
        TXI      BGINO4,1,-1                                 FINP0184
        TXI BGINO1,1,-1                                      FINP0185
BGINO1  LXA M6B35,2            6                             FINP0186
        TSX OCTAL,4                                          FINP0187
        TXH 4                                                FINP0188
BGINO2  TSX SGINOO,4                                         FINP0189
BGINO3  TXL GIN11,,**          EXIT FOR NEW PREFIX           FINP0190
BGINO4  TXH      BGINO3,2,5       BLANK                      FINP0191
        TXL      BGINO2                                      FINP0192
IGINOO  TSX DECNO,4           INTERGER FORMAT                FINP0193
        TXH GINER,2,0              ERROR FOR NON-INTEGER     FINP0194
        LLS      18                                          FINP0195
        TXL BGINO2                                           FINP0196
DGINOO  LDQ 0,1                                              FINP0197
        TSX SGINOO,4                                         FINP0198
        CLA MASK2                                            FINP0199
        STO CRDADD                                           FINP0200
        LDQ 1,1                                              FINP0201
```

```
        TXI BGINO2,1,-1                                     FINP0202
DECNO   SXD BGINO3,4                                        FINP0203
        CLA 1,1                                             FINP0204
        ALS 24                                              FINP0205
        SLW COMMON+20                                       FINP0206
        STZ SUM                                             FINP0207
        STZ SIGN                                            FINP0208
        LXA 100B35,2                                        FINP0209
        CLS M6B35           6                               FINP0210
        TSX DECIM,4                                         FINP0211
        TXI DECN2+1,1,-1    END ON BLANK                    FINP0212
        TXI DECN1,1,-1                                      FINP0213
DECN1   CLA 4B35            4                               FINP0214
        TSX DECIM,4                                         FINP0215
DECN2   TXH 2,,**                                           FINP0216
        TXL DECN4,2,99                                      FINP0217
        TXL GIN11,2,100                                     FINP0218
DECN3   LXD 1B35,2                                          FINP0219
DECN4   LXD BGINO3,4                                        FINP0220
        LXD BGINO3,4                                        FINP0221
        TRA 1,4                                             FINP0222
DECEN   SXD GINO3,4                                         FINP0223
        TSX DECNO,4                                         FINP0224
        STQ COMMON+13                                       FINP0225
        SXD DECN2,2                                         FINP0226
        CLA DECN2                                           FINP0227
        STZ SUM                                             FINP0228
        STZ SIGN                                            FINP0229
        LDQ COMMON+20                                       FINP0230
        STQ 0,1                                             FINP0231
        TSX DECIM,4                                         FINP0232
        TXH                                                 FINP0233
        LXD GINO3,4                                         FINP0234
        TRA 1,4                                             FINP0235
FGINOO  TSX DECEN,4     FIXED PT. CONVERSION                FINP0236
        STQ COMMON                                          FINP0237
        CAL DECN2                                           FINP0238
        COM                                                 FINP0239
        ADD 1B17                                            FINP0240
        PDX 0,2                                             FINP0241
        CLA COMMON                                          FINP0242
        SUB BREF,2                                          FINP0243
        TMI GINER                                           FINP0244
        STA FGINO1                                          FINP0245
        LDQ COMMON+13                                       FINP0246
        MPY FREF,2                                          FINP0247
FGINO1  LRS **                                              FINP0248
        TNZ GINER                                           FINP0249
        TXL BGINO2,,**                                      FINP0250
LGINOO  TSX DECEN,4     FLTG. PT. CONVERSION                FINP0251
        STQ COMMON                                          FINP0252
```

```
       LXD DECN2,2                                    FINP0253
       PXD ,2                                         FINP0254
       ARS 18                                         FINP0255
       SSM                                            FINP0256
       ADD COMMON                                     FINP0257
       LRS 35                                         FINP0258
       DVP 10B35                                      FINP0259
       SUB M9B35                                      FINP0260
       PAX ,2                                         FINP0261
       CLM                                            FINP0262
       LLS 35                                         FINP0263
       ADD 5B35           HIGH DIGIT                  FINP0264
       TMI GINER                                      FINP0265
       PAX ,4                                         FINP0266
       SXD FGINO1+2,1     SAVE ADD REFERENCE          FINP0267
       CLA XREF1,2                                    FINP0268
       ADD XREF2,4                                    FINP0269
       ADD 126B35                                     FINP0270
       TPL LGINOA                                     FINP0271
       STZ COMMON+13                                  FINP0272
LGINOA PAX 0,1                                        FINP0273
       LDQ FREF1,2                                    FINP0274
       MPR FREF2,4                                    FINP0275
       LRS 35                                         FINP0276
       MPY COMMON+13                                  FINP0277
       LLS 2                                          FINP0278
       TZE LGINO3                                     FINP0279
LGINO1 TZE LGINO2                                     FINP0280
       LRS 1                                          FINP0281
       TXI LGINO1,1,1                                 FINP0282
LGINO2 PXD ,1                                         FINP0283
       ARS 18                                         FINP0284
       LLS 0                                          FINP0285
       LRS 8                                          FINP0286
LGINO3 LXD FGINO1+2,1                                 FINP0287
       TNZ GINER                                      FINP0288
       TXL BGINO2         EXIT TO STORE               FINP0289
SGINOO CLA CRDADD         STORE ROUTINE               FINP0290
       SUB MASK2                                      FINP0291
       TZE SGINO1                                     FINP0292
       CLA ADDIND                                     FINP0293
       SUB MASK3                                      FINP0294
       TZE SGINOA                                     FINP0295
       CLA CRDADD                                     FINP0296
       TXL SGINO2                                     FINP0297
SGINOA CAL     GINOO                                  FINP0298
       SUB     CRDADD                                 FINP0299
       TXL SGINO2                                     FINP0300
SGINO1 CAL SGINO3         ADD 1 TO OLD ADDRESS        FINP0301
       SUB     1B35                                   FINP0302
SGINO2 STA SGINO3                                     FINP0303
```

```
SGINO3 STQ **
       TRA 1,4
OCTAL  LDQ 0,1                   SET-UP OCTAL INTEGER
OCTAL1 PXD
       LGL 3
       TNZ OCTAL2
       CAL SUM
       LGL 3
       SLW SUM
       TIX OCTAL1,2,1
       TXI OCTAL3,4,-1
OCTAL2 LGL 3
       SUB MASK3
       TNZ GINER
OCTAL3 LDQ SUM
       TRA 1,4
DECIM  SXD DECIM2,4              GENERATE DECIMAL INTEGER
       PAX ,4
DECIM1 CLM
       LDQ 0,1
       LGL 6
       STQ 0,1
       SUB 10B35
       TMI DECIM6
       SUB 17B35
       TNZ DECIM3
       LXD 10B35,2
DECIM2 TXL DECIM7,,**
DECIM3 SUB 5B35
       TMI GINER
       SUB 10B35
       TNZ DECIM4
       SUB 10B35
DECIM4 TPL DECIM5
       STO SIGN
       TXL DECIM6
DECIM5 ADD M6B35
       TNZ GINER
DECIM8 LXD DECIM2,4              END ON BLANK
       LDQ SUM
       CLA SIGN
       LRS 0
       TRA 1,4
DECIM6 ADD 10B35
       STO COMMON+14
       LDQ SUM
       MPY 10B35
       LLS 35
       ADD COMMON+14
       STO SUM
       TXI DECIM7,2,1
```

```
DECIM7 TIX DECIM1,4,1                                    FINP0355
       LXD DECIM2,4                                      FINP0356
       TXI DECIM8+1,4,-1     END ON DIGIT COUNT          FINP0357
       DEC 133                                           FINP0358
100B35 DEC 100                                           FINP0359
       DEC 67                                            FINP0360
       DEC 34                                            FINP0361
1B35   DEC 1                                             FINP0362
       DEC -33                                           FINP0363
       DEC -66                                           FINP0364
       DEC -99                                           FINP0365
       DEC -132                                          FINP0366
XREF2  DEC .9183549616B0                                 FINP0367
       DEC .7888609053B0                                 FINP0368
       DEC .6776263579B0                                 FINP0369
       DEC .5820766092B0                                 FINP0370
       DEC .5B0                                          FINP0371
       DEC .8589934592B0                                 FINP0372
       DEC .7378697630B0                                 FINP0373
       DEC .6338253002B0                                 FINP0374
       DEC .5444517871B0                                 FINP0375
FREF2  DEC 30                                            FINP0376
       DEC 27                                            FINP0377
       DEC 24                                            FINP0378
       DEC 20                                            FINP0379
17B35  DEC 17                                            FINP0380
       DEC 14                                            FINP0381
10B35  DEC 10                                            FINP0382
       DEC 7                                             FINP0383
4B35   DEC 4                                             FINP0384
BREF   DEC 1                                             FINP0385
       DEC -3                                            FINP0386
M6B35  DEC -6                                            FINP0387
M9B35  DEC -9                                            FINP0388
       DEC -13                                           FINP0389
       DEC -16                                           FINP0390
       DEC -19                                           FINP0391
       DEC -23                                           FINP0392
       DEC -26                                           FINP0393
XREF1  DEC -29                                           FINP0394
       DEC .9313225747B0                                 FINP0395
       DEC .7450580597B0                                 FINP0396
       DEC .5960464478B0                                 FINP0397
       DEC .9536743165B0                                 FINP0398
       DEC .7629394531B0                                 FINP0399
       DEC .6103515625B0                                 FINP0400
       DEC .9765625B0                                    FINP0401
       DEC .78125B0                                      FINP0402
       DEC .625B0                                        FINP0403
FREF   DEC .5B0                                          FINP0404
       DEC .8B0                                          FINP0405
```

```
        DEC .64B0                                          FINP040
        DEC .512B0                                         FINP040
        DEC .8192B0                                        FINP040
        DEC .65536B0                                       FINP040
        DEC .524288B0                                      FINP041
        DEC .8388608B0                                     FINP041
        DEC .67108864B0                                    FINP041
FREF1   DEC .536870912B0                                   FINP041
5B35    DEC 5                                              FINP041
126B35  PZE 126                                            FINP041
MASK1   OCT 000000000077                                   FINP041
MASK2   OCT 606060606000                                   FINP041
MASK3   OCT 000000000060                                   FINP041
1B17    OCT 1000000                                        FINP041
 2B35   DEC 2B35                                           FINP042
 3B35   PZE 3                                              FINP042
 8B35   DEC 8B35                                           FINP042
11B35   PZE      11                                        FINP042
12B35   PZE 12                                             FINP042
MTX00   SXA      MTX10,4         TSX MTX00,4               FINP042
        AXT      2,4             IJ RETURN                 FINP042
MTX01   SXA      MTX02,4         NORMAL RETURN             FINP042
        STZ      SIGN                                      FINP042
        STZ      SUM                                       FINP042
        LXA      100B35,2                                  FINP043
        CLA      2B35                                      FINP043
        TSX      DECIM,4                                   FINP043
        TXH      0,0,0                                     FINP043
        TXL      GINER,2,100                               FINP043
MTX02   AXT      **,4                                      FINP043
        STQ      I+2,4           I,J                       FINP043
        LDQ      0,1             REMOVE                    FINP043
        LGL      6                  COMMA                  FINP043
        STQ      0,1                                       FINP043
        TIX      MTX01,4,1                                 FINP044
MTX10   AXT      **,4                                      FINP044
        CLA      J                                         FINP044
        SUB      1B35                                      FINP044
        XCA                                                FINP044
        MPY      IMAX                                      FINP044
        XCA                                                FINP044
        ADD      I                                         FINP044
        LDQ      IJ                                        FINP044
        TLQ      1,4             LOC GTR MAX               FINP044
        SUB      1B35                                      FINP045
        XCA                                                FINP045
        TRA      2,4                                       FINP045
*       SET UP IJ WORD                                     FINP045
MCD00   SXA      MCD02,4                                   FINP045
        CAL      GIN00                                     FINP045
        ADD      1B35                                      FINP045
```

```
         STA     SGIN03                                      FINP0457
         TSX     MTX00,4                                     FINP0458
MCD02    NOP     **                                          FINP0459
         LDQ     I                                           FINP0460
         STQ     IMAX                                        FINP0461
         MPY     J                                           FINP0462
         STQ     IJ                                          FINP0463
         LXA     MCD02,4                                     FINP0464
         TXL     MCD04,4,12                                  FINP0465
         LXA     IJ,4                                        FINP0466
         TXI     *+1,4,-1                                    FINP0467
         SXD     *+6,4                                       FINP0468
         CLA     GIN00                                       FINP0469
         STA     *+2                                         FINP0470
         LXD     1B35,4                                      FINP0471
         STZ     **,4                                        FINP0472
         TXI     *+1,4,1                                     FINP0473
         TXL     *-2,4,**                                    FINP0474
MCD04    TXI     GIN11,1,-1                                  FINP0475
*                                                            FINP0476
ACD0     STZ     CTR                                         FINP0477
         STZ     PASS                                        FINP0478
ACD1     STZ     PF                                          FINP0479
         AXT     5,2                                         FINP0480
         TSX     SYM00,4                                     FINP0481
         AXT     0,4             BLANK                       FINP0482
         TXH     ACD6,2,1                                    FINP0483
         TXL     ACD12,4,0       NULL AND BLANK              FINP0484
         TXL     ACD6+1          NULL                        FINP0485
ACD6     STQ     PF                                          FINP0486
         TXL     ACD10,4,0       BLANK                       FINP0487
         AXT     1,2                                         FINP0488
         TSX     SYM00,4                                     FINP0489
         AXT     0,4             BLANK                       FINP0490
         TXL     ACD10,2,1       NULL                        FINP0491
         XCL                                                 FINP0492
         ALS     15                                          FINP0493
         STT     PF                                          FINP0494
ACD10    LDQ     PF                                          FINP0495
         CLS     PASS                                        FINP0496
         STO     PASS                                        FINP0497
         TPL     ACD15                                       FINP0498
         STQ     WORD                                        FINP0499
         TXH     ACD1,4,0        NOT BLANK                   FINP0500
ACD12    LXA     CTR,4                                       FINP0501
         TXH     ACD14,4,6                                   FINP0502
         TXI     *+1,1,-1                                    FINP0503
         TXL     *+3,4,1         BLANK                       FINP0504
ACD14    LDQ     WORD                                        FINP0505
         TSX     SGIN00,4                                    FINP0506
         TXL     GIN11                                       FINP0507
```

```
ACD15  RQL     18                                                FINP05
       SLQ     WORD                                              FINP05
       TXL     ACD12                                             FINP05
*                                                                FINP05
SYM00  SXD     SYM02,2          LXA L(NO CHARACTERS),2           FINP05
       SXA     SYM14,4          TSX SYM00,4                      FINP05
       STZ     PSYM             BALNK                            FINP05
       AXT     1,2              NORMAL IR2=1 IF NULL             FINP05
SYM01  CLA     CTR                                               FINP05
       ADD     1B35                                              FINP05
       STO     CTR                                               FINP05
       SUB     10B35+1          7                                FINP05
       TNZ     *+2                                               FINP05
       TXI     *+1,1,-1                                          FINP05
       LDQ     0,1                                               FINP05
       CAL     PSYM                                              FINP05
       LGL     6                                                 FINP05
       SLW     PSYM                                              FINP05
       STQ     0,1                                               FINP05
       ANA     MASK1                                             FINP05
       SUB     MASK3                                             FINP05
       TZE     SYM20            BLANK                            FINP05
       SUB     11B35                                             FINP05
       TZE     SYM21            COMMA                            FINP05
       TXI     *+1,2,1                                           FINP05
SYM02  TXL     SYM01,2,**                                        FINP05
       CAL     PSYM                                              FINP05
SYM03  TXI     *+1,2,-1                                          FINP05
       SXA     SYM06,2          NO CHARACTERS                    FINP05
       LDQ     MASK2                                             FINP05
SYM04  LGL     6                LEFT JUSTIFY                     FINP05
       TXI     *+1,2,1                                           FINP05
       TXL     SYM04,2,5                                         FINP05
       SLW     SYMBL                                             FINP05
       CAL     PSYM                                              FINP05
       ANA     BLANK                                             FINP05
       TNZ     SYM10            SYMBOLIC                         FINP05
SYM06  AXT     **,2             NO CHARACTERS                    FINP05
       LDQ     SYMBL                                             FINP05
       STZ     SUM                                               FINP05
       STZ     SIGN                                              FINP05
       TXL     SYM08,2,4        DECIMAL                          FINP05
       TSX     OCTAL1,4         OCTAL                            FINP05
       TXL     GINER                                             FINP05
       TSX     SYM14,2                                           FINP05
SYM08  CLA     0,1                                               FINP05
       STQ     0,1                                               FINP05
       STO     SYMBL                                             FINP05
       PXA     ,2                                                FINP05
       TSX     DECIM,4                                           FINP05
       TXL     GINER                                             FINP05
```

```
        CLA     SYMBL                                          FINP0559
        STO     0,1                                            FINP0560
        TSX     SYM14,2                                        FINP0561
SYM10   CLA     SYMBL                                          FINP0562
        TSX     TBL00,4                                        FINP0563
SYM14   AXT     **,4                                           FINP0564
        TRA     2,4                  EXIT                      FINP0565
SYM20   TXI     *+1,4,1              BLANK                     FINP0566
        SXA     SYM14,4                                        FINP0567
SYM21   TXL     SYM14+1,2,1          NULL FIELD                FINP0568
SYM25   CAL     PSYM                                           FINP0569
        ARS     6                                              FINP0570
        SLW     PSYM                                           FINP0571
        TXL     SYM03                                          FINP0572
TBL00   LXD     TBL01+1,2                                      FINP0573
TBL01   CAS     **,2                                           FINP0574
        TXL     *+2                                            FINP0575
        TXI     TBL02,2,-2                                     FINP0576
        TXI     *+1,2,-1                                       FINP0577
TBL03   TXH     TBL01,2,**                                     FINP0578
        TXL     GINER,0,-1                                     FINP0579
TBL02   SXA     *+1,2                                          FINP0580
        AXC     0,2                                            FINP0581
        SXA     *+2,2                                          FINP0582
        LXD     FINPT,2                                        FINP0583
        LDQ     **,2                                           FINP0584
        LXD     TBL03+1,2                                      FINP0585
        TRA     1,4                                            FINP0586
IMAX    PZE                                                    FINP0587
IJ      PZE                                                    FINP0588
COMMON  BSS     31                                             FINP0589
GIN00   SYN     COMMON+30                                      FINP0590
PF      SYN     COMMON+15                                      FINP0591
PASS    SYN     COMMON+16                                      FINP0592
CTR     EQU     COMMON+17                                      FINP0593
SYMBL   EQU     COMMON+18                                      FINP0594
PSYM    EQU     COMMON+19                                      FINP0595
CRDADD  EQU     COMMON+21                                      FINP0596
SIGN    EQU     COMMON+22                                      FINP0597
SUM     EQU     COMMON+23                                      FINP0598
ADDIND  EQU     COMMON+24                                      FINP0599
PREFIX  EQU     COMMON+25                                      FINP0600
WORD    SYN     COMMON+26                                      FINP0601
ERROR   SYN     COMMON+27                                      FINP0602
I       SYN     PASS                                           FINP0603
J       SYN     CTR                                            FINP0604
FINPT   SYN     FINP                                           FINP0605
        END                                                    FINP
```

Function: FNORM

Purpose: To find the magnitude of a 3-dimensional vector.

Calling Sequence:

Y = FNORM(X)

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	X	3	X_1, X_2, X_3		
O	FNORM	1			$FNORM = (X_1^2 + X_2^2 + X_3^2)^{\frac{1}{2}}$

Common storages used or required: None

Subroutines required: None

Functions required: SQRTF

Approximate number of storages required:

```
*      SYMBOL TABLE                                                   FNOR
CLC2018  FUNCTION FNORM                                               FNOR
       FUNCTION FNORM(X)                                              FNOR0000
       DIMENSION X(3)                                                 FNOR0010
     1 FNORM = SQRTF(X(1)**2+X(2)**2+X(3)**2)                         FNOR0020
     3 RETURN                                                         FNOR0030
       END                                                            FNOR
```

Subroutine: GHA

Purpose: To determine the Greenwich Hour Angle of the first point of Aries for a given date and time.

Calling Sequence:

CALL GHA (TSEC, D, GHAN, DA, OMEGA)

Input and Output

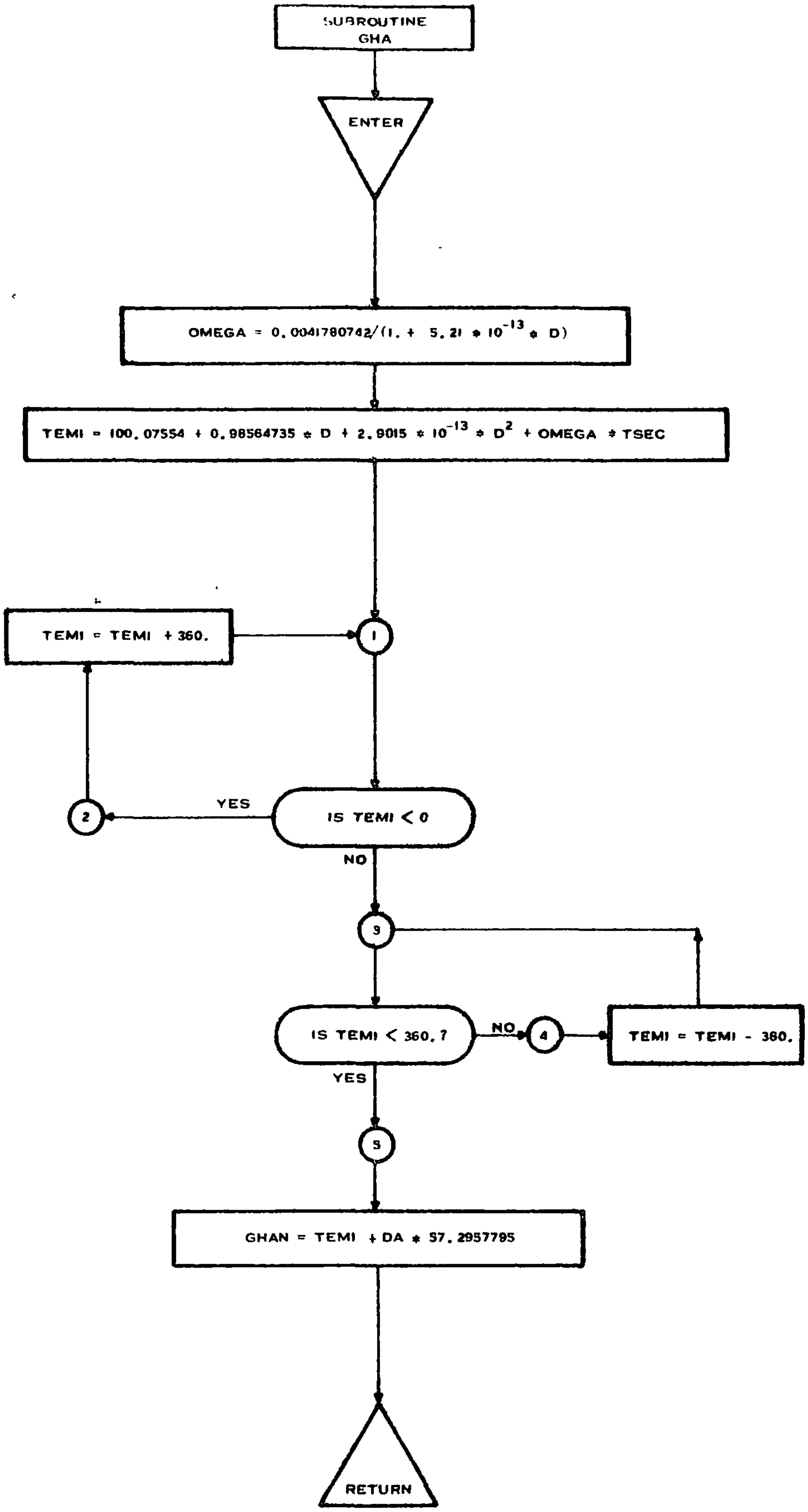
I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	TSEC	1		seconds	Fractional part of day from D
I	D	1		days	Whole days from Jan. 1, 1950
O	GHAN	1		degrees	Greenwich Hour Angle,
					$0 \leq GHA < 360$
I	DA	1			Adjustment due to nutation*
O	OMEGA	1	ω_e	degrees/sec	Rate of rotation of the earth
					*DA = EN(2,1) of EN nutation
					matrix from NUTATE

Common storages used or required: None

Subroutines required: None

Functions required: None

Approximate number of storages required: ____


SUBROUTINE GHA
ENTER
OMEGA = 0.0041780742/(1. + 5.21 * 10^-13 * D)
TEMI = 100.07554 + 0.98564735 * D + 2.9015 * 10^-13 * D^2 + OMEGA * TSEC
TEMI = TEMI + 360.
1
YES
2
IS TEMI < 0
NO
3
IS TEMI < 360. ?
NO
4
TEMI = TEMI - 360.
YES
5
GHAN = TEMI + DA * 57.2957795
RETURN

GHA

```
*       LABEL
CEC20AG
        SUBROUTINE GHA(TSEC,D,GHAN,DA,OMEGA)
        OMEGA = .0041780742/(1.+5.21E-13*D)
        DD=D
D       DD=DD*(.98564735/360.)
D       DD=DD-INTF(DD)
        DF=DD
        TEM1 = 100.07554+360.*DF     +2.9015E-13*D*D+OMEGA*TSEC
      1 IF (TEM1) 2,3,3
      2 TEM1 = TEM1+360.
        GO TO 1
      3 IF (TEM1-360.) 5,4,4
      4 TEM1 = TEM1-360.
        GO TO 3
      5 GHAN = TEM1+DA*57.2957795
        RETURN
        END
```

Page Intentionally Left Blank

Subroutine: GOTOB

Purpose: It is a logic type of subroutine to read input data cards, set up a sequence for calling subroutine MATSUB, interpret types of measurements being made, and control integration package to perform specific operations (read new data, call MATSUB, stop, etc.) at specified times.

Calling Sequence:

CALL GOTOB (TSTPD)

Input and Output

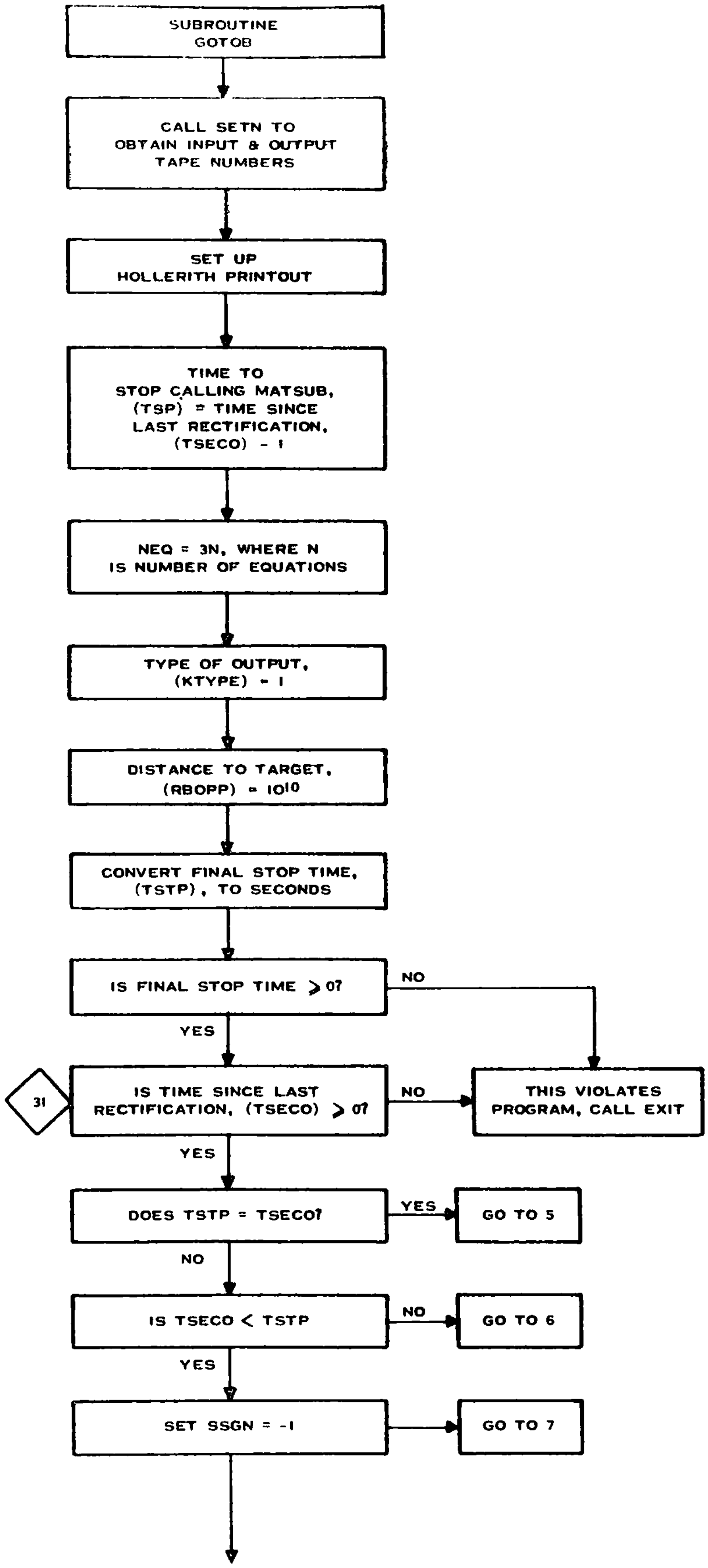
I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	TSTPD	1	time	days	Desired stop time in days from start

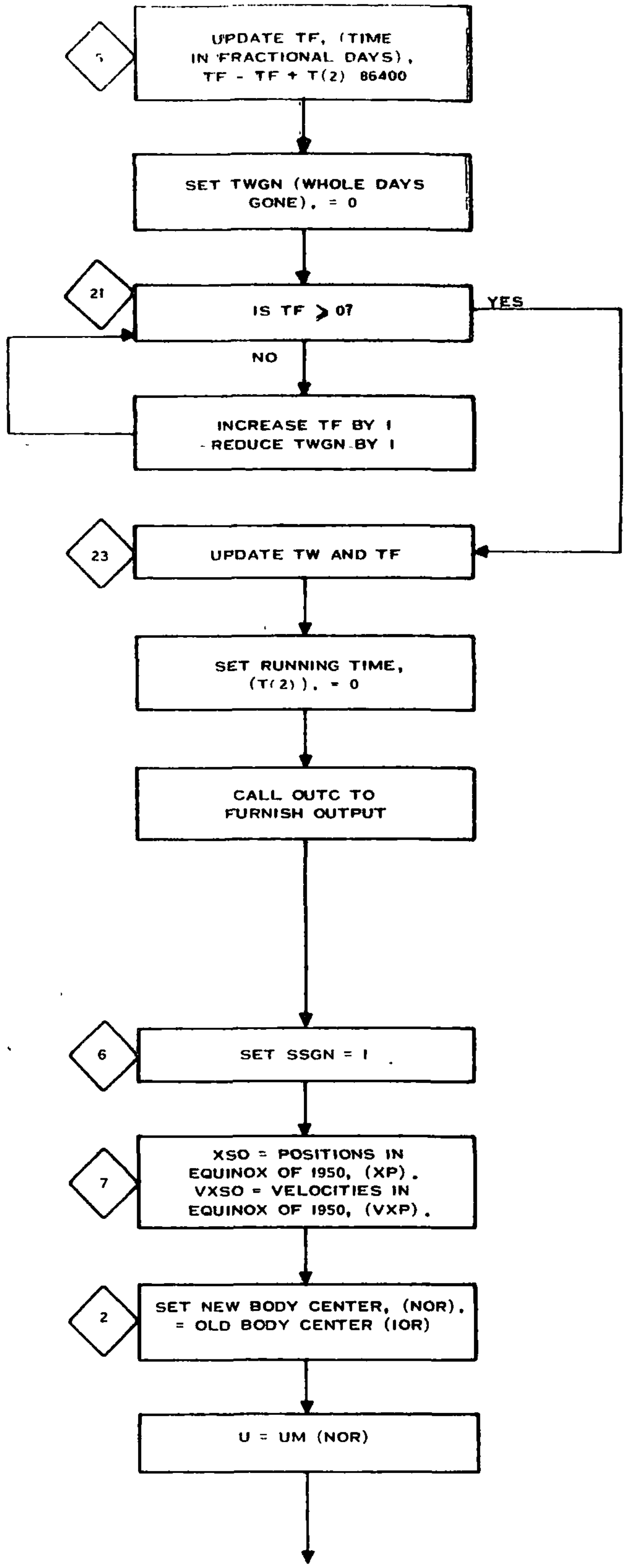
Common storages used or required: T, S, C, IC

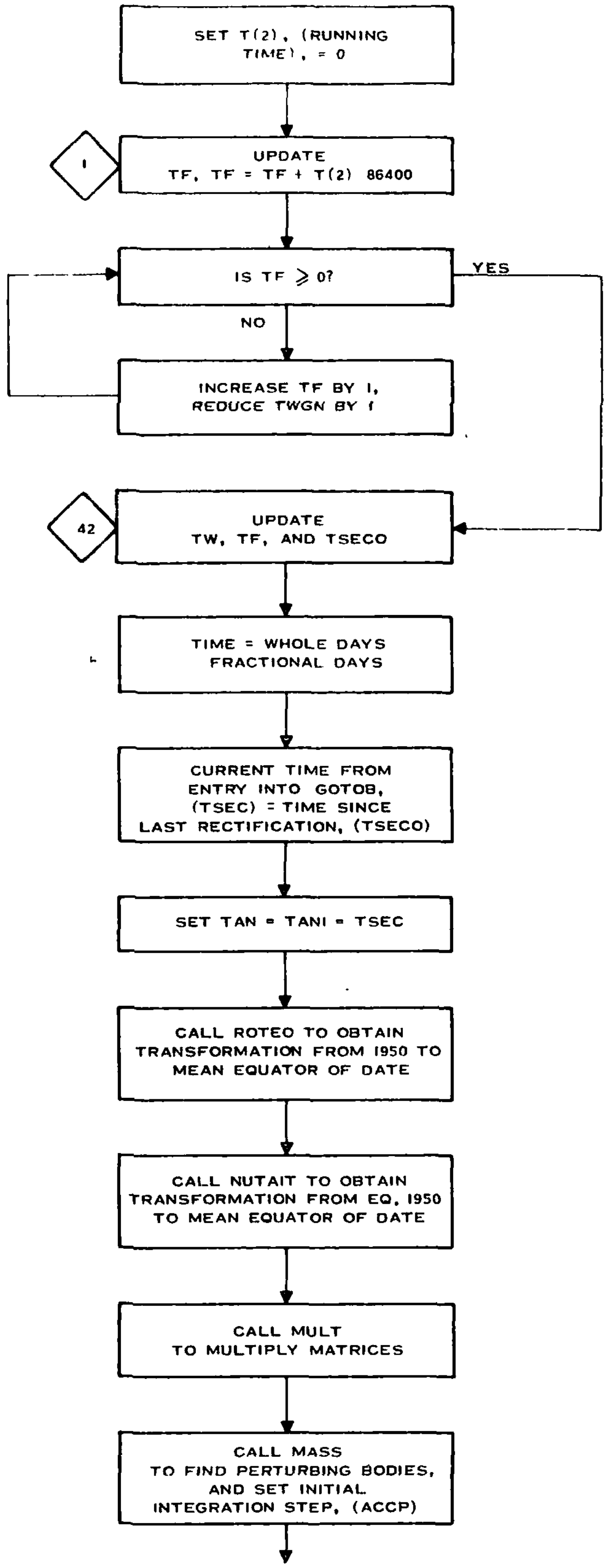
Subroutines required: CHNGP, DE6FN, DE6FP, DE6FP3, DE6FP4, EXIT, MASS, MATSUB, MULT, NUTAIT, OUTC, ROTEQ, SDEC, SETN, SHIFTP, TIMED

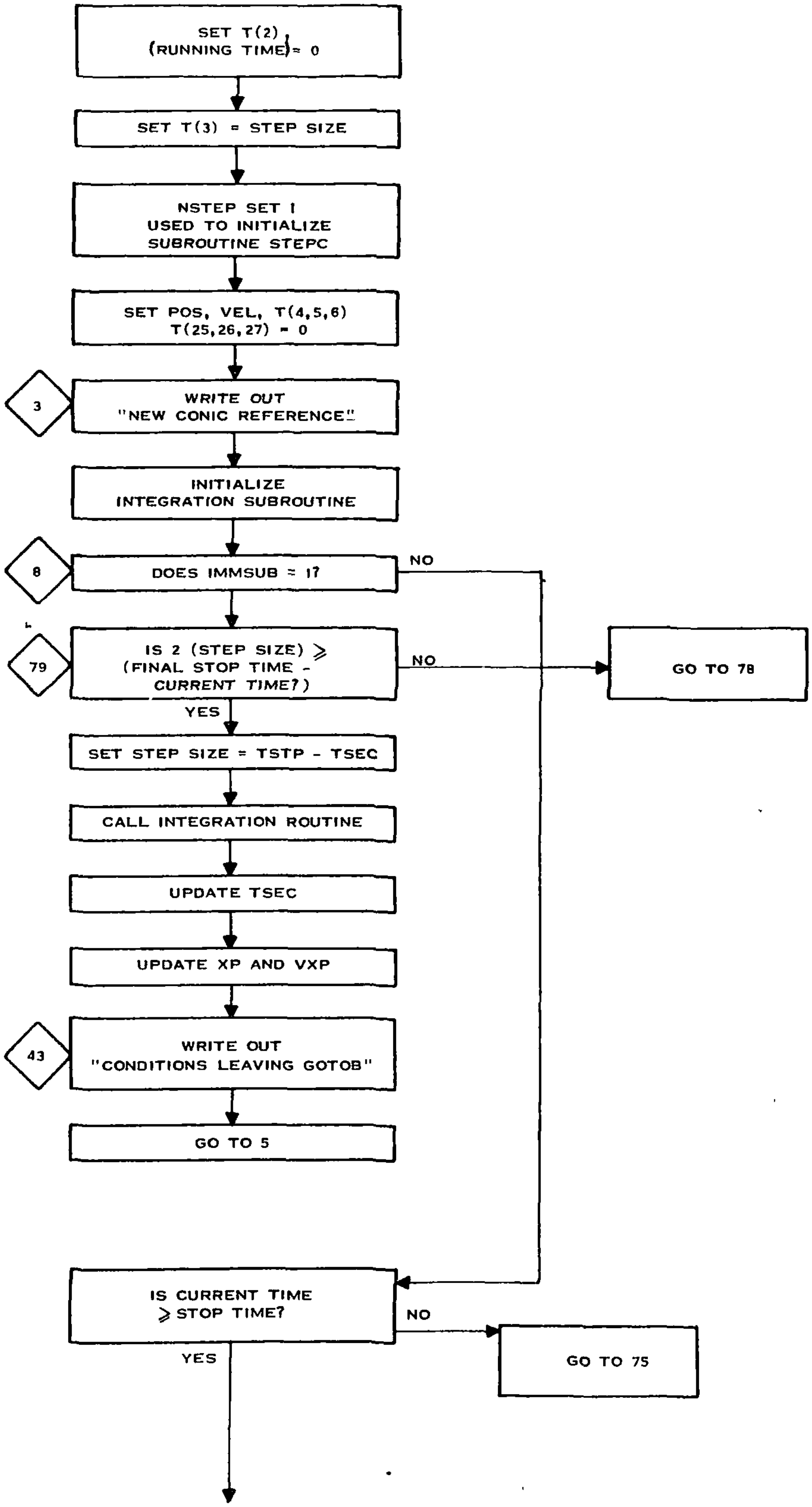
Functions required: FNORM (FIL)(RTN)(SLI)

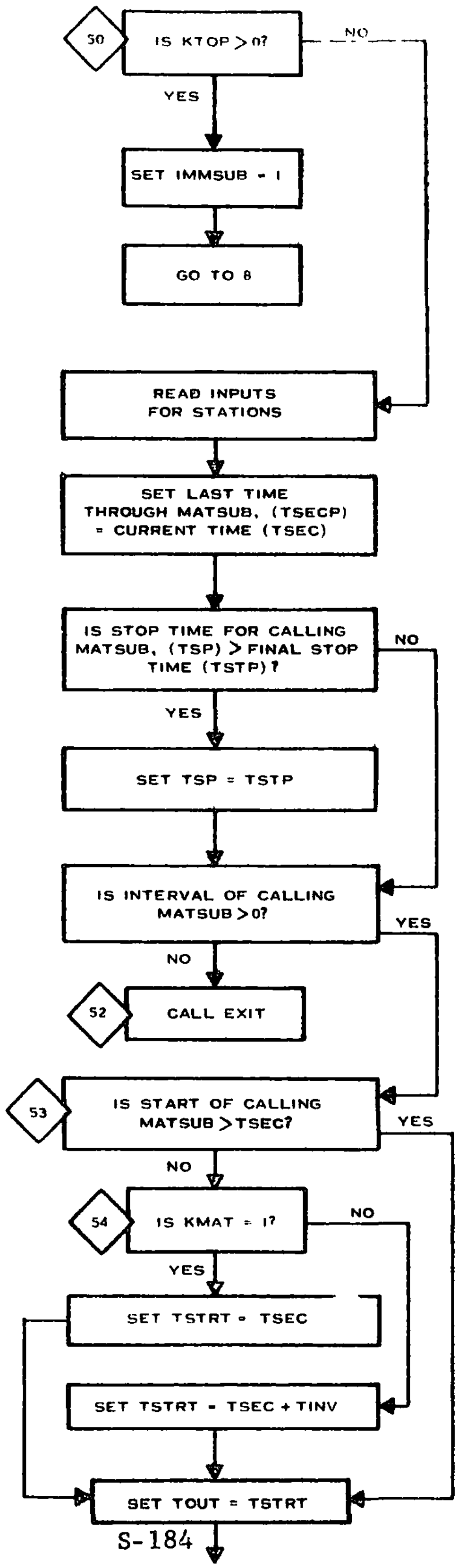
Approximate number of storages required: 1242 DEC

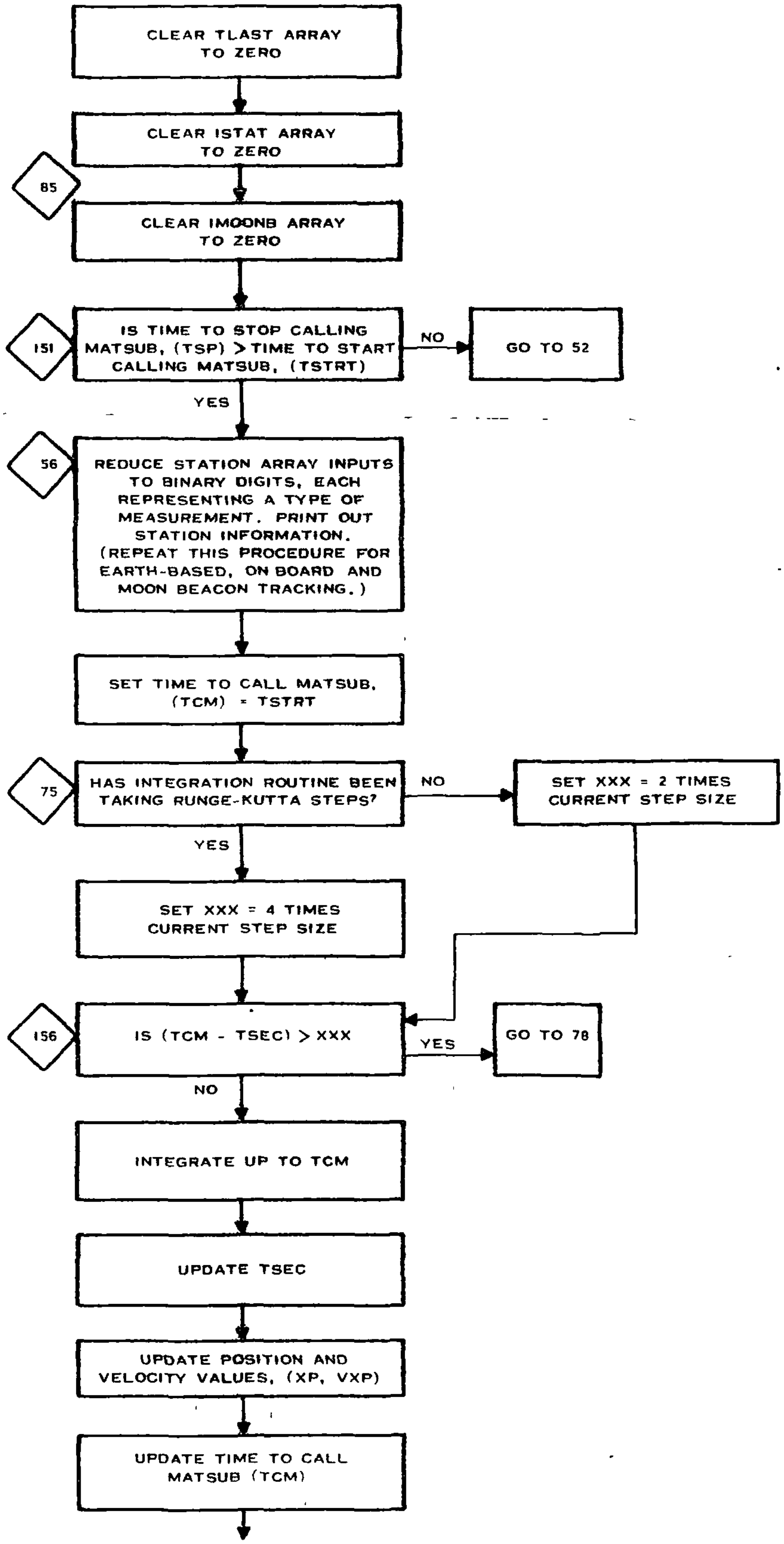

SUBROUTINE GOTOB
CALL SETN TO OBTAIN INPUT & OUTPUT TAPE NUMBERS
SET UP HOLLERITH PRINTOUT
TIME TO STOP CALLING MATSUB, (TSP) = TIME SINCE LAST RECTIFICATION, (TSECO) - 1
NEQ = 3N, WHERE N IS NUMBER OF EQUATIONS
TYPE OF OUTPUT, (KTYPE) = 1
DISTANCE TO TARGET, (RBOPP) = 10^{10}
CONVERT FINAL STOP TIME, (TSTP), TO SECONDS
IS FINAL STOP TIME > 0?
NO
YES
31
IS TIME SINCE LAST RECTIFICATION, (TSECO) > 0?
NO
THIS VIOLATES PROGRAM, CALL EXIT
YES
DOES TSTP = TSECO?
YES
GO TO 5
NO
IS TSECO < TSTP
NO
GO TO 6
YES
SET SSGN = -1
GO TO 7

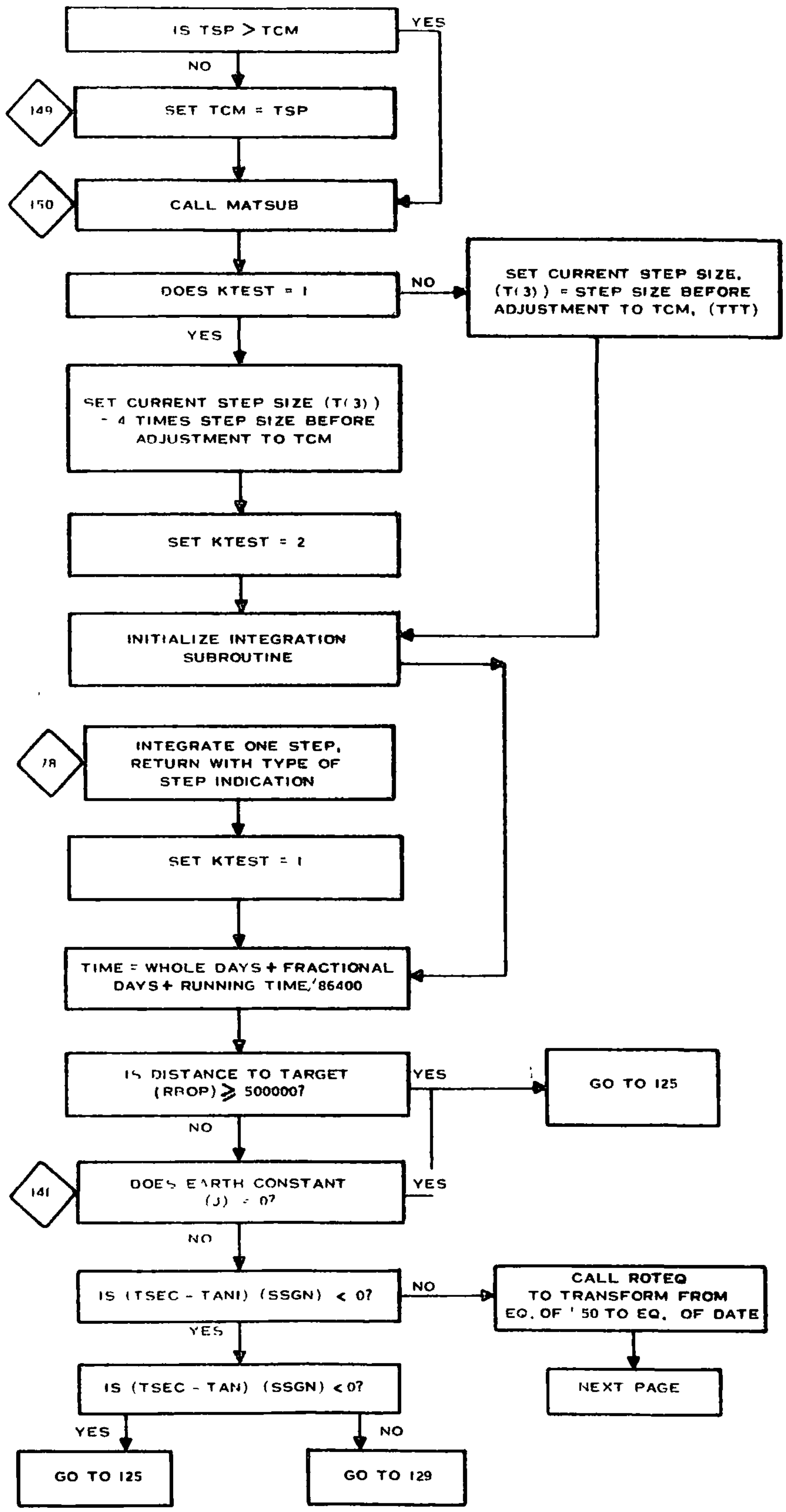

UPDATE TF, (TIME IN FRACTIONAL DAYS), TF - TF + T(2) 86400
SET TWGN (WHOLE DAYS GONE), = 0
21
IS TF > 0?
YES
NO
INCREASE TF BY 1 REDUCE TWGN BY 1
23
UPDATE TW AND TF
SET RUNNING TIME, (T(2)), = 0
CALL OUTC TO FURNISH OUTPUT
6
SET SSGN = 1
7
XSO = POSITIONS IN EQUINOX OF 1950, (XP). VXSO = VELOCITIES IN EQUINOX OF 1950, (VXP).
2
SET NEW BODY CENTER, (NOR), = OLD BODY CENTER (IOR)
U = UM (NOR)

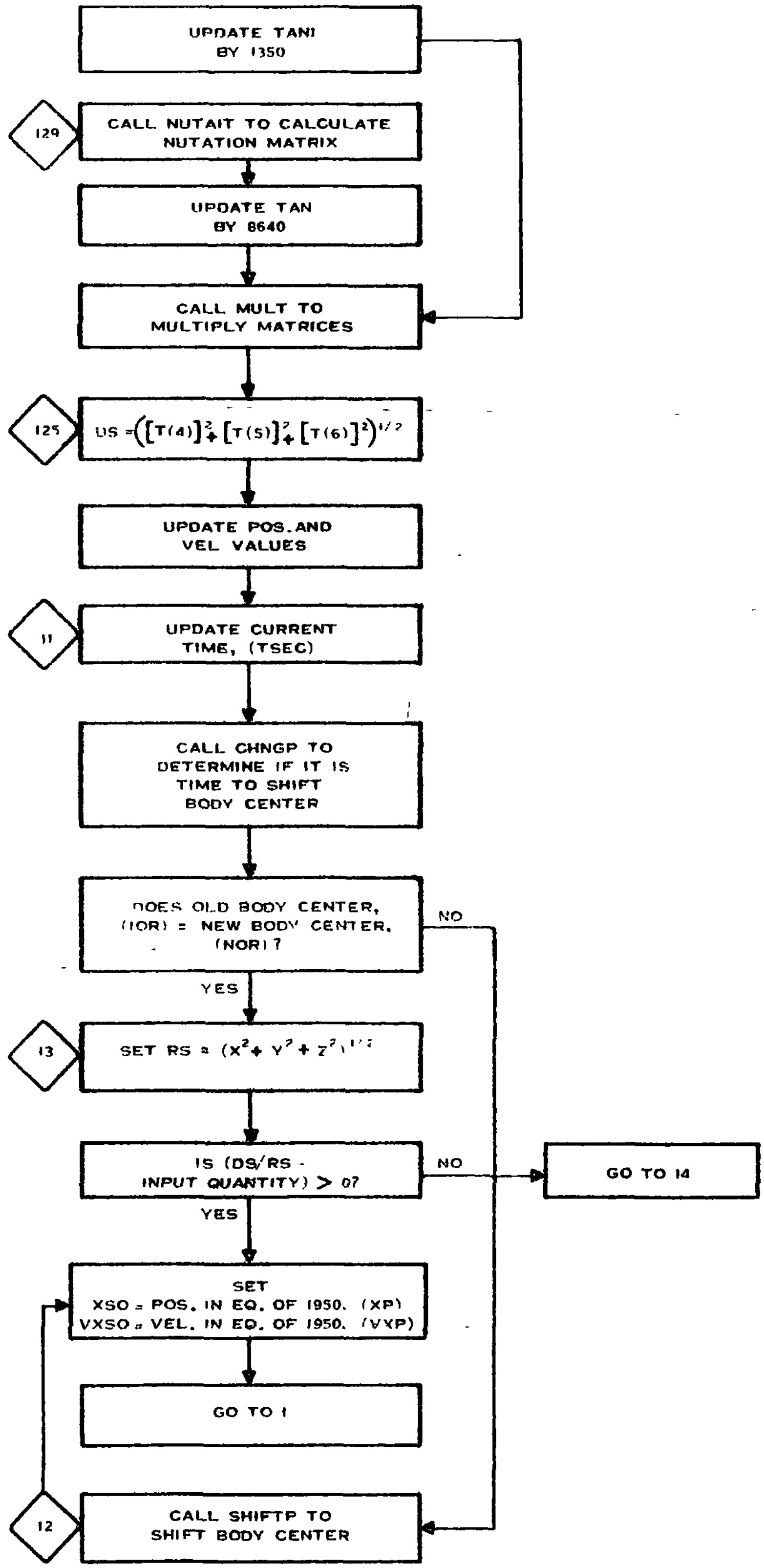

SET T(2), (RUNNING TIME), = 0
1
UPDATE TF, TF = TF + T(2) 86400
IS TF ⩾ 0?
YES
NO
INCREASE TF BY 1, REDUCE TWGN BY 1
42
UPDATE TW, TF, AND TSECO
TIME = WHOLE DAYS FRACTIONAL DAYS
CURRENT TIME FROM ENTRY INTO GOTOB, (TSEC) = TIME SINCE LAST RECTIFICATION, (TSECO)
SET TAN = TANI = TSEC
CALL ROTEO TO OBTAIN TRANSFORMATION FROM 1950 TO MEAN EQUATOR OF DATE
CALL NUTAIT TO OBTAIN TRANSFORMATION FROM EQ. 1950 TO MEAN EQUATOR OF DATE
CALL MULT TO MULTIPLY MATRICES
CALL MASS TO FIND PERTURBING BODIES, AND SET INITIAL INTEGRATION STEP, (ACCP)


SET T(2) (RUNNING TIME) = 0
SET T(3) = STEP SIZE
NSTEP SET 1 USED TO INITIALIZE SUBROUTINE STEPC
SET POS, VEL, T(4,5,6) T(25,26,27) = 0
3
WRITE OUT "NEW CONIC REFERENCE"
INITIALIZE INTEGRATION SUBROUTINE
8
DOES IMMSUB = 1?
NO
79
IS 2 (STEP SIZE) ≥ (FINAL STOP TIME - CURRENT TIME?)
NO
GO TO 78
YES
SET STEP SIZE = TSTP - TSEC
CALL INTEGRATION ROUTINE
UPDATE TSEC
UPDATE XP AND VXP
43
WRITE OUT "CONDITIONS LEAVING GOTOB"
GO TO 5
IS CURRENT TIME ≥ STOP TIME?
NO
GO TO 75
YES

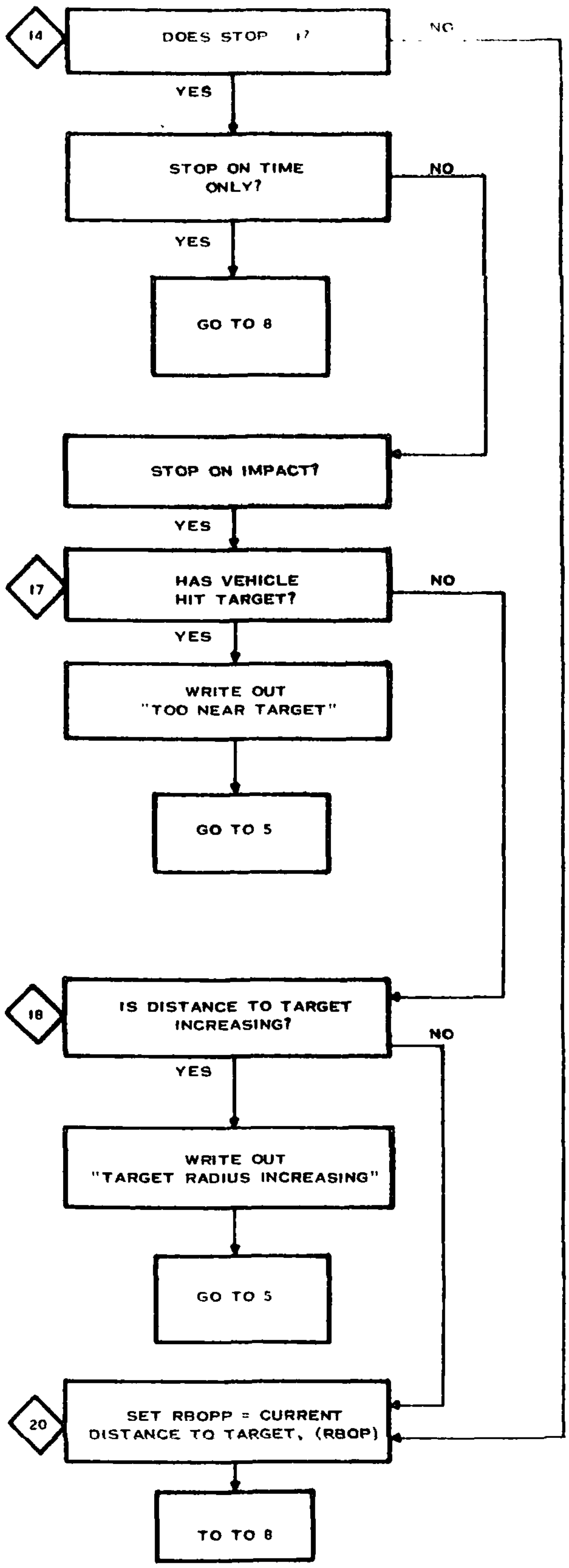

50
IS KTOP > 0?
NO
YES
SET IMMSUB = 1
GO TO 8
READ INPUTS FOR STATIONS
SET LAST TIME THROUGH MATSUB, (TSECP) = CURRENT TIME (TSEC)
IS STOP TIME FOR CALLING MATSUB, (TSP) > FINAL STOP TIME (TSTP)?
NO
YES
SET TSP = TSTP
IS INTERVAL OF CALLING MATSUB > 0?
YES
NO
52
CALL EXIT
53
IS START OF CALLING MATSUB > TSEC?
YES
NO
54
IS KMAT = 1?
NO
YES
SET TSTRT = TSEC
SET TSTRT = TSEC + TINV
SET TOUT = TSTRT


CLEAR TLAST ARRAY TO ZERO
CLEAR ISTAT ARRAY TO ZERO
85
CLEAR IMOONB ARRAY TO ZERO
151
IS TIME TO STOP CALLING MATSUB, (TSP) > TIME TO START CALLING MATSUB, (TSTRT)
NO
GO TO 52
YES
56
REDUCE STATION ARRAY INPUTS TO BINARY DIGITS, EACH REPRESENTING A TYPE OF MEASUREMENT. PRINT OUT STATION INFORMATION. (REPEAT THIS PROCEDURE FOR EARTH-BASED, ON BOARD AND MOON BEACON TRACKING.)
SET TIME TO CALL MATSUB, (TCM) = TSTRT
75
HAS INTEGRATION ROUTINE BEEN TAKING RUNGE-KUTTA STEPS?
NO
SET XXX = 2 TIMES CURRENT STEP SIZE
YES
SET XXX = 4 TIMES CURRENT STEP SIZE
156
IS (TCM - TSEC) > XXX
YES
GO TO 78
NO
INTEGRATE UP TO TCM
UPDATE TSEC
UPDATE POSITION AND VELOCITY VALUES, (XP, VXP)
UPDATE TIME TO CALL MATSUB (TCM)


IS TSP > TCM
YES
NO
149
SET TCM = TSP
150
CALL MATSUB
DOES KTEST = 1
NO
SET CURRENT STEP SIZE, (T(3)) = STEP SIZE BEFORE ADJUSTMENT TO TCM, (TTT)
YES
SET CURRENT STEP SIZE (T(3)) = 4 TIMES STEP SIZE BEFORE ADJUSTMENT TO TCM
SET KTEST = 2
INITIALIZE INTEGRATION SUBROUTINE
78
INTEGRATE ONE STEP, RETURN WITH TYPE OF STEP INDICATION
SET KTEST = 1
TIME = WHOLE DAYS + FRACTIONAL DAYS + RUNNING TIME/86400
IS DISTANCE TO TARGET (RPOP) ≥ 5000000?
YES
GO TO 125
NO
141
DOES EARTH CONSTANT (J) = 0?
YES
NO
IS (TSEC - TANI) (SSGN) < 0?
NO
CALL ROTEQ TO TRANSFORM FROM EQ. OF '50 TO EQ. OF DATE
NEXT PAGE
YES
IS (TSEC - TAN) (SSGN) < 0?
YES
NO
GO TO 125
GO TO 129



UPDATE TANI BY 1350
129
CALL NUTAIT TO CALCULATE NUTATION MATRIX
UPDATE TAN BY 8640
CALL MULT TO MULTIPLY MATRICES
125
US = ([T(4)]² + [T(5)]² + [T(6)]²)^1/2
UPDATE POS. AND VEL VALUES
11
UPDATE CURRENT TIME, (TSEC)
CALL CHNGP TO DETERMINE IF IT IS TIME TO SHIFT BODY CENTER
DOES OLD BODY CENTER, (IOR) = NEW BODY CENTER, (NOR)?
NO
YES
13
SET RS = (X² + Y² + Z²)^1/2
IS (DS/RS - INPUT QUANTITY) > 0?
NO
GO TO 14
YES
SET
XSO = POS. IN EQ. OF 1950. (XP)
VXSO = VEL. IN EQ. OF 1950. (VXP)
GO TO 1
12
CALL SHIFTP TO SHIFT BODY CENTER


14
DOES STOP ?
NO
YES
STOP ON TIME ONLY?
NO
YES
GO TO 8
STOP ON IMPACT?
YES
17
HAS VEHICLE HIT TARGET?
NO
YES
WRITE OUT "TOO NEAR TARGET"
GO TO 5
18
IS DISTANCE TO TARGET INCREASING?
NO
YES
WRITE OUT "TARGET RADIUS INCREASING"
GO TO 5
20
SET RBOPP = CURRENT DISTANCE TO TARGET, (RBOP)
TO TO 8

GOTOB-10

GOTOB

```
*       LABEL                                                                    GOTB
*       SYMBOL TABLE                                                             GOTB
CEC2004  SUBROUTINE GOTOB                                                        GOTB
      SUBROUTINE GOTOB(TSTPD)                                                    GOTB00
      DIMENSION T(1360),S(1000),C(1000),IC(300),VX(3)                            GOTB00
      DIMENSION VXP(3),XP(3),AE(3),AO(3),AP(3),PO(22),VE(22),CA(3)               GOTB00
     1,X(3),UM(6),RAD(6),RBOP(6),                          IMOONB(10)            GOTB00
     1,VKB(6),RBO(6),AN(3,3),EN(3,3),EA(3,3)                                     GOTB00
     1,XSO(3),VXSO(3)                                                            GOTB00
      DIMENSION BP(3,3),TLAST(4,6)                                               GOTB00
      DIMENSION ISTAT(20),IONB(6),IMB(10),KS(4,20),KONB(4,6),KMB(4,10)           GOTB00
     1,ISA(20)                                                                   GOTB00
      DIMENSION SMEAS(4),BNAME(6),BEACNO(10),TOBM(4)                             GOTB00
      COMMON T,S,C,IC                                                            GOTB00
      EQUIVALENCE (T,TDUM),(S,SDUM),(C,CDUM)                                     GOTB01
      EQUIVALENCE (S(1),UM),(S(11),VJ),(S(12),H),(S(13),D),(S(14),RE)            GOTB01
     1,(S(900),RAD),(C(9),SSGN),(C(10),TW),(C(11),TF),(C(12),TSECO)              GOTB01
     2,(C(13),TP1),(C(14),TP2),(C(15),XP),(C(18),VXP),(C(21),AE)                 GOTB01
     3,(C(24),AO),(C(27),CA),(C(30),TSEC),(C(33),X),(C(36),VX)                   GOTB01
     5,(C(46),XSO),(C(49),VXSO),        (C(55),U),(C(56),VKB),(C(62),PO)         GOTB01
     6,(C(84),VE),(C(106),RBO),(C(112),RBOP),(C(120),EN),(C(129),EA)             GOTB01
     7,(C(138),AN),(C(171),BP)                                                   GOTB01
     8,(IC(1),N),(IC(2),IOR),(IC(3),NOR),(IC(4),KSTP),(IC(5),KTYPE)              GOTB01
     9,(IC(6),ITARG),(IC(7),KOUT),(IC(220),IMOONB)                               GOTB02
      EQUIVALENCE (IC(10),ISA),(IC(110),IONB),(IC(140),IMB),(IC(30),KS)          GOTB02
     1,(IC(116),KONB),(IC(150),KMB),(C(650),TPRINT),(IC(9),LPRINT)               GOTB02
     2,(C(651),TSTRT),(IC(190),LSTAT),(IC(191),LONB),(IC(192),LMB)               GOTB02
     3,(IC(8),IMMSUB),(C(622),TLAST),(C(649),TSECP),(IC(194),ISTAT)              GOTB02
     4,(C(647),TOUT)          ,(IC(231),NSTEP),(S(485),ENKE)                     GOTB02
      CALL SETN(NIN,NOUT)                                                        GOTB02
C SMEAS MEANS TYPE OF STATION MEASUREMENT                                         GOTB02
      SMEAS(1) = 6HRANGE.                                                        GOTB02
      SMEAS(2)= 6HAZ-ELE                                                         GOTB02
      SMEAS(3)= 6HDIRCOS                                                         GOTB03
      SMEAS(4) = 6HRNGRAT                                                        GOTB03
C BNAME MEANS BODY NAME                                                           GOTB03
      BNAME(1) = 6HEARTH.                                                        GOTB03
      BNAME(2) = 6HMOON..                                                        GOTB03
      BNAME(3) = 6HSUN...                                                        GOTB03
      BNAME(4) = 6HVENUS.                                                        GOTB03
      BNAME(5) = 6HMARS..                                                        GOTB03
      BNAME(6) = 6HJUPITR                                                        GOTB03
C TOBM MEANS TYPE OF ONBOARD MEASUREMENT                                          GOTB03
      TOBM(1) = 6HRANGE.                                                         GOTB04
      TOBM(2) = 6HRTASCN                                                         GOTB04
      TOBM(3) = 6HDECLIN                                                         GOTB04
      TOBM(4) = 6HRNGRAT                                                         GOTB04
      KTEST=2                                                                    GOTB04
      TSP =TSECO-1.                                                              GOTB04
      LPRINT=1                                                                   GOTB04
      KMAT=1                                                                     GOTB04
```

(Cont'd)

```
      K=I+NEQV                                                          GOTB1320
      XP(I)=X(I)+T(I+3)                                                 GOTB1330
      VXP(I)=VX(I)+T(K)                                                 GOTB1340
   43 CONTINUE                                                          GOTB1350
      WRITE OUTPUT TAPE NOUT,39                                         GOTB1360
   39 FORMAT(25H0CONDITIONS LEAVING GOTOB)                              GOTB1370
      WRITE OUTPUT TAPE NOUT,750                                        GOTB1380
  750 FORMAT(21H STOPPED ON TIME STOP)                                  GOTB1390
      GO TO 5                                                           GOTB1400
   10 CONTINUE                                                          GOTB1410
      IF(TSEC-TSP)75,50,50                                              GOTB1420
   50 CONTINUE                                                          GOTB1430
C     KTOP IS SET +1 ON LAST CARD OF SEQUENCE FOR NORMAL TIME STOP      GOTB1435
C     OF GOTOB                                                          GOTB1436
      IF(KTOP)51,51,80                                                  GOTB1440
   80 CONTINUE                                                          GOTB1450
      IMMSUB=1                                                          GOTB1460
      GO TO 8                                                           GOTB1470
   51 READ INPUT TAPE NIN,700,ISA,IONB,IMB                              GOTB1480
  700 FORMAT(20I2,4X,6I2,4X,10I2)                                       GOTB1490
      READ INPUT TAPE NIN,701,KTOP,TSTR,TSP1,TINT,TPRIN                 GOTB1500
  701 FORMAT(1X,I1,4E17.8)                                              GOTB1510
      WRITE OUTPUT TAPE NOUT,708,KTOP,TSTR,TSP1,TINT,TPRIN              GOTB1520
  708 FORMAT(/7H0 KTOP=,I1,7H  TSTR=,1E17.8,7H  TSP1=,1E17.8,7H  TINT=  GOTB1530
     1,1E17.8,8H  TPRIN=,1E17.8)                                        GOTB1540
      TSECP=TSEC                                                        GOTB1550
C     SUBROUTINE TIMED CONVERTS INPUT TIMES FROM FORMAT                 GOTB1555
C     (DAYS HOURS).(MIN SEC)                                            GOTB1556
      CALL TIMED(TSTR,TSTRT)                                            GOTB1560
C     TSTRT =START TIME FOR CALLING MATSUB                              GOTB1570
      CALL TIMED(TSP1,TSP)                                              GOTB1580
C     TSP   =STOP TIME AND INDICATOR FOR NEW DATA                       GOTB1590
      STOP=KTOP                                                         GOTB1600
      IF(TSP-TSTP)101,101,100                                           GOTB1610
  100 TSP=TSTP                                                          GOTB1620
  101 CONTINUE                                                          GOTB1630
      CALL TIMED(TINT,TINV)                                             GOTB1640
C     TINV  =INTERVAL OF CALLING MATSUB                                 GOTB1650
      CALL TIMED(TPRIN,TPRINT)                                          GOTB1660
C     TPRINT=PRINT INTERVAL FOR MATSUB                                  GOTB1670
      IF(TINV) 52,52,53                                                 GOTB1680
   52 WRITE OUTPUT TAPE NOUT,702                                        GOTB1690
  702 FORMAT(26H0YOU BOTCHED UP INPUT DATA)                             GOTB1700
      CALL EXIT                                                         GOTB1710
   53 CONTINUE                                                          GOTB1720
      IF(TSTRT-TSEC) 54,54,158                                          GOTB1730
   54 CONTINUE                                                          GOTB1740
      GO TO (152,153),KMAT                                              GOTB1750
  152 TSTRT=TSEC                                                        GOTB1760
      TOUT=TSEC                                                         GOTB1770
      GO TO 158                                                         GOTB1780
```

GOTOB-(Con't)

```
  153 TSTRT=TSEC+TINV
      TOUT=TSEC+TPRINT
      GO TO 151
 158  CONTINUE
      TOUT=TSTRT
      DO 84 I=1,4
      DO 84 J=1,6
      TLAST(I,J) =0.
   84 CONTINUE
      DO 85 I=1,20
      ISTAT(I) =0
  85  CONTINUE
      DO 86 I=1,10
      IMOONB(I)=0,
  86  CONTINUE
  151 CONTINUE
      IF(TSP-TSTRT)  52,52,55
   55 CONTINUE
      LSTAT=1
      LONB=1
      LMB=1
      L=122
      DO 61 I=1,20
      L=L+15
      IF (ISA(I)) 61,61,56
   56 CONTINUE
      WRITE OUTPUT TAPE NOUT,709,I,S(L)
  709 FORMAT (17H  STATION NUMBER ,I2,2H ,,A6,2H ,,10H  MEASURES)
      BBB = ISA(I)
      LSTAT=2
      DO 60 J=1,4
      CCC=BBB/2.
      DDD=INTF(CCC)
      IF(CCC-DDD) 57,57,58
   57 CONTINUE
C     THE  KS, KONB, AND KMB ARRAYS ARE  COLUMN WISE ARRANGED IN BINARY
C     FOR INDICATING TYPE MEASUREMENTS MADE .KS FOR EARTH BASED TRACKING
C     KONB FOR ONBOARD TRACKING, KMB FOR LUNAR BEACONS.
      KS(J,I)=0
      GO TO 59
   58 CONTINUE
      KS(J,I)=1
      WRITE OUTPUT TAPE NOUT,703,SMEAS(J)
  703 FORMAT (1H A6)
   59 CONTINUE
      BBB=DDD
   60 CONTINUE
   61 CONTINUE
      DO 62 I=1,6
      IF (IONB(I))  62,62,63
   63 CONTINUE
```

GOTOB (Cont'd)

```
      WRITE OUTPUT TAPE NOUT,704,BNAME(I)
  704 FORMAT (28H  VEHICLE TO CELESTIAL BODY ,A6,10H  MEASURES)
      LONB=2
      BBB=IONB(I)
      DO  68 J=1,4
      CCC=BBB/2.
      DDD=INTF(CCC)
      IF(CCC-DDD) 64,64,65
   64 CONTINUE
      KONB(J,I)=0
      GO TO 67
   65 CONTINUE
      KONB(J,I)=1
      WRITE OUTPUT TAPE NOUT,703,TOBM(J)
   67 CONTINUE
      BBB=DDD
   68 CONTINUE
   62 CONTINUE
      DO  74 I=1,10
      IF (IMB(I))  74,74,69
   69 CONTINUE
      WRITE OUTPUT TAPE NOUT,706,I
  706 FORMAT (33H  VEHICLE TO LUNAR BEACON NUMBER ,I2,10H  MEASURES)
      LMB=2
      BBB=IMB(I)
      DO  73 J=1,4
      CCC=BBB/2.
      DDD=INTF(CCC)
      IF(CCC-DDD) 70,70,71
   70 CONTINUE
      KMB(J,I)=0
      GO TO 72
   71 CONTINUE
      KMB(J,I)=1
      WRITE OUTPUT TAPE NOUT,703,TOBM(J)
   72 CONTINUE
      BBB=DDD
   73 CONTINUE
   74 CONTINUE
      TCM=TSTRT
   75 CONTINUE
      IF(ACCUM) 154,154,155
  155 XXX=4.*T(3)
      GO TO 156
  154 XXX=2.*T(3)
  156 CONTINUE
      IF(TCM-TSEC-XXX) 76,76,78
   76 CONTINUE
      TT=TCM-TSEC
      TTT=T(3)
      T(3)=TT
```

GOTOB (Cont'd)

```
      DO 77 I=1,4                                                       GOTB281
      CALL DE6FP3(TT)                                                   GOTB282
   77 CONTINUE                                                          GOTB283
      TSEC = TSECO + T(2)                                               GOTB284
      DO 157 I=1,3                                                      GOTB285
      K=I+NEQV                                                          GOTB286
      XP(I) = X(I)+T(I+3)                                               GOTB287
  157 VXP(I) = VX(I)+T(K)                                               GOTB288
      TCM=TCM+TINV                                                      GOTB289
      IF(TCM-TSP) 150,150,149                                           GOTB290
  149 TCM=TSP                                                           GOTB291
  150 CONTINUE                                                          GOTB292
      CALL MATSUB                                                       GOTB293
      KMAT=2                                                            GOTB294
      IF(KTEST-1)81,81,82                                               GOTB295
  81  T(3)=4.*TTT                                                       GOTB296
      KTEST=2                                                           GOTB297
      GO TO 87                                                          GOTB298
  82  T(3)=TTT                                                          GOTB298
  87  CONTINUE                                                          GOTB299
      CALL DE6FN(-1,1,T,NEQ,SDEC,3,4,1.E-9,3.456,172800.,86400.)        GOTB300
      GO TO 113                                                         GOTB301
   78 CONTINUE                                                          GOTB302
      CALL DE6FP1(ACCUM)                                                GOTB303
      KTEST=1                                                           GOTB303
  113 TIME = TW+TF + T(2)/86400.                                        GOTB304
      IF (RBOP-500000.) 141,125,125                                     GOTB305
  141 IF (VJ) 127,125,127                                               GOTB306
  127 IF((TSEC-TAN1)*SSGN)142,126,126                                   GOTB307
  142 IF((TSEC-TAN)*SSGN)125,129,129                                    GOTB308
  126 CALL ROTEQ(TIME,EA)                                               GOTB309
      TAN1=TAN1+1350.*SSGN                                              GOTB310
      GO TO 140                                                         GOTB311
  129 CALL NUTAIT (TIME,CM,CRUD,DDC,EN,EPSIL)                           GOTB312
      TAN=TAN+8640.*SSGN                                                GOTB313
  140 CALL MULT(EN,EA,AN,0)                                             GOTB314
  125 CONTINUE                                                          GOTB315
      DS=FNORM(T(4))                                                    GOTB316
      DO 11 I=1,3                                                       GOTB317
      K=I+NEQV                                                          GOTB318
      XP(I)=X(I)+T(I+3)                                                 GOTB319
      VXP(I)=VX(I)+T(K)                                                 GOTB320
   11 CONTINUE                                                          GOTB321
      TSEC = TSECO + T(2)                                               GOTB322
      CALL CHNGP(NOR,IOR,RBOP)                                          GOTB323
      IF(IOR-NOR)12,13,12                                               GOTB324
   13 CONTINUE                                                          GOTB325
      RS=FNORM(X)                                                       GOTB326
      IF(DS/RS-ENKE)14,15,15                                            GOTB327
   15 CONTINUE                                                          GOTB328
      DO 16 I=1,3                                                       GOTB329
```

GOTOB (Cont'd)

```
      XSO(I)=XP(I)                                                      GOTB3300
      VXSO(I)=VXP(I)                                                    GOTB3310
   16 CONTINUE                                                          GOTB3320
      GO TO 1                                                           GOTB3330
   12 CONTINUE                                                          GOTB3340
      CALL SHIFTP(IOR,NOR,U,UM,XP,VXP,PO,VE,TF,TW,T(2))                 GOTB3350
      GO TO 15                                                          GOTB3360
   14 CONTINUE                                                          GOTB3370
      ITARG = ITARG                                                     GOTB3380
      IF(STOP-1.)20,83,83                                               GOTB3390
   83 CONTINUE                                                          GOTB3400
      KSTP=KSTP                                                         GOTB3410
      GO TO (17,18,8),KSTP                                              GOTB3420
   17 CONTINUE                                                          GOTB3430
      IF(RBOP(ITARG)-RAD(ITARG))19,19,18                                GOTB3440
   19 CONTINUE                                                          GOTB3450
      WRITE OUTPUT TAPE NOUT,39                                         GOTB3460
      WRITE OUTPUT TAPE NOUT,34,RBOP(ITARG),RAD(ITARG),ITARG            GOTB3470
   34 FORMAT(21H TOO NEAR TARG  RBOP=,1E17.8,5H RAD=,1E17.8,6H TARG=,I1)GOTB3480
      KTYPE=2                                                           GOTB3490
      GO TO 5                                                           GOTB3500
   18 CONTINUE                                                          GOTB3510
      IF(RBOP(ITARG)-RBOPP)20,35,35                                     GOTB3520
   35 CONTINUE                                                          GOTB3530
      WRITE OUTPUT TAPE NOUT,39                                         GOTB3540
      WRITE OUTPUT TAPE NOUT,36,RBOP(ITARG),RBOPP,ITARG                 GOTB3550
   36 FORMAT(29H TARG RADIUS INCREASING RBOP=,1E17.8,7H RBOPP=,1E17.8,6HGOTB3560
     1 TARG=,I1)                                                        GOTB3570
      KTYPE=2                                                           GOTB3580
      GO TO 5                                                           GOTB3590
   20 CONTINUE                                                          GOTB3600
      RBOPP=RBOP(ITARG)                                                 GOTB3610
      GO TO 8                                                           GOTB3620
      END                                                               GOTB
```

Subroutine: GOTOR

Purpose: To solve Kepler's equation for incremental excentric anomaly on a conic section given the incremental mean anomaly (time).

Calling Sequence:

CALL GOTOR (K, VM, C, F, E1)

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	K	1	K		Orbit type (1) Elliptic
					(2) Hyperbolic
I	VM	1	ΔM	RAD	Incremental mean anomaly
I	C	2	$C_1 C_2$		Kepler's equation coefficients
O	F	4	F		Trigonometric functions of
					incremental eccentric anomaly
I/O		1	ΔE	RAD	Incremental eccentric anomaly

Common storages used or required: None

Subroutines required: None

Functions required: SIN, COS, EXP

Approximate number of storages required: 476 DEC

Discussion (See subroutine STEPC)

The incremental eccentric anomaly, φ, is implicitly expressed as a function of incremental mean anomaly, ΔM, and conic coefficients, C_1 and C_2, by transcendental equations

$$\Delta M = (\varphi - \sin\varphi) + C_1 \sin\varphi - C_2(\cos\varphi - 1) \qquad \underline{\text{elliptical}}$$

$$\Delta M = (\sinh\varphi - \varphi) + C_1 \sinh\varphi + C_2(\cosh\varphi - 1) \qquad \underline{\text{hyperbolic}}$$

The two equations above may be written as a single equation

$$\Delta M = f_1(\varphi) + C_1 f_3(\varphi) + C_2 f_4(\varphi)$$

if the convention below is adopted.

For the elliptical case, let

$$f_1(\varphi) = \varphi - \sin\varphi$$

$$f_2(\varphi) = 1 - \cos\varphi$$

$$f_3(\varphi) = \sin\varphi$$

$$f_4(\varphi) = \cos\varphi$$

and for the hyperbolic case, let

$$f_1(\varphi) = \sinh\varphi - \varphi$$

$$f_2(\varphi) = \cosh\varphi - 1$$

$$f_3(\varphi) = \sinh\varphi$$

$$f_4(\varphi) = \cosh\varphi$$

When $|\varphi| \leq 1$, numerical accuracy requires that $f_1(\varphi)$ and $f_2(\varphi)$ be computed by truncated series expansions. Otherwise, the computer library functions SINF, COSF and EXPF are used in the computation.

The slope of the function

$$F(\varphi) = f_1(\varphi) + C_1 f_3(\varphi) + C_2 f_4(\varphi)$$

at φ is seen to be

$$F'(\varphi) = f_2(\varphi) + C_1 f_4(\varphi) + C_2 f_3(\varphi).$$

$F(\varphi)$ is a monotonically increasing function of φ so that any solution of the equation $F(\varphi) = \Delta M$ is obviously unique. Newton's method of iteration is used. That is, letting φ_n be the n^{th} estimate of φ, the (n+1)st estimate is calculated from

$$\varphi_{n+1} = \varphi_n + \frac{\Delta M - F(\varphi_n)}{F'(\varphi_n)}$$

The iteration is halted and φ is said to be φ_n when

$$\left| \frac{\varphi_{n+1} - \varphi_n}{\varphi_{n+1} + \varphi_n} \right| \leq 3. \times 10^{-8}$$

or when n = 20, whichever occurs first. The $f_i(\varphi)$ as well as φ are output from GOTOR to avoid their re-computation outside the subroutine.

GOTOR

```
*     LABEL
CEC2091  NEW GOTOR
      SUBROUTINE GOTOR(K,VM,C,F,E1)
      DIMENSION C(2),F(4)
      NMAX=20
      N=0
      GO TO (1,2),K
    1 CONTINUE
C     FIRST GUESS IS OBTAINED FOR ELLIPTICAL CASE
    8 CONTINUE
      IF(E1-1.)30,31,31
   30 CONTINUE
      D2=E1*E1
      F(1)=E1*D2*(0.16666667E+00-D2*(0.833333333E-02-D2*(0.198412698E-03
     1-D2*(0.275573192E-05-D2*0.250521083E-07))))
      F(2)= D2*(0.5-D2*(0.41666667E-1-D2*(0.13888889E-2-D2*(0.24801587E-
     14-D2*0.27557319E-6))))
      F(3)=E1-F(1)
      F(4)=1.-F(2)
      GO TO 3
   31 CONTINUE
      F(3)=SINF(E1)
      F(4)=COSF(E1)
      F(1)=E1-F(3)
      F(2)=1.-F(4)
      GO TO 3
    2 CONTINUE
C     FIRST GUESS IS OBTAINED FORHYPERBOLIC CASE
    9 CONTINUE
      IF(E1-1.)32,33,33
   32 CONTINUE
      D2=E1*E1
      F(1) =E1*D2*(0.16666667E+00+D2*(0.83333333E-02+D2*(0.198412698E-03
     1+D2*(0.275573192E-05+D2*0.250521083E-07))))
      F(2)  =D2*(0.5+D2*(0.41666667E-1+D2*(0.13888889E-2+D2*(0.24801587E
     1-4+D2*0.27557319E-6))))
      F(3)=E1+F(1)
      F(4)=1.+F(2)
      GO TO 3
   33 CONTINUE
      EX=.5*EXPF(E1)
      OX=.25/EX
      F(3)=EX-OX
      F(4)=EX+OX
      F(1)=F(3)-E1
      F(2)=F(4)-1.
    3 CONTINUE
      CM=F(1)+C(1)*F(3)+C(2)*F(2)
      DM=F(2)+C(1)*F(4)+C(2)*F(3)
      DE=(VM-CM)/DM
      ERROR=DE
```

GOTOR(Cont'd)

```
      AB=ABSF(DE)                                                       GOTR050
      IF(AB-1.)10,10,11                                                 GOTR051
   11 DE=DE/AB                                                          GOTR052
   10 E2=E1+DE                                                          GOTR053
      IF(ABSF((E2-E1)/(E2+E1))-3.E-8)4,4,5                              GOTR054
    5 CONTINUE                                                          GOTR055
      IF(N-NMAX)6,7,7                                                   GOTR056
    7 CONTINUE                                                          GOTR057
      GO TO 4                                                           GOTR058
    6 CONTINUE                                                          GOTR059
      N=N+1                                                             GOTR060
      E1=E2                                                             GOTR061
      GO TO (8,9),K                                                     GOTR062
    4 CONTINUE                                                          GOTR063
      RETURN                                                            GOTR064
      END                                                               GOTR
```

Page Intentionally Left Blank

Subroutine: GUID

Purpose: To determine the times when guidance corrections are to be made. Compute the correction to be made and determine the error in the correction being performed. Then the subroutine updates both the state and guidance covariance matricies for the correction.

Calling Sequence:

CALL GUID

Input and **Output**

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I					Quantities obtained from common
O					Quantities returned to common

Common storages used or required: T, S, C, IC

Subroutines required: INV3, MARX, SETN, SQRTF, TIMED

Functions required: (FIL)(STH)

Approximate number of storages required: 1022 DEC

Theory for updating covariance matricies for a guidance correction.

The following quantities will be used in the derivation to follow.

$PAR(to)$ = covariance matrix of errors at injection

$\hat{x}(t)$ = estimate of state at time t

$\tilde{x}(t)$ = error in estimate of state at time t

$x(t)$ = true state at time t

$E(\tilde{x}\,\tilde{x}^T) = P(t)$ = covariance matrix of error in estimate at time t

The quantities $\hat{x}$, $\tilde{x}$, and x are all considered as deviation quantities from the nominal trajectory which arrives at the target in the desired manner.

Using linear theory for the deviations

$$x(t) = \Phi\, X(to)$$

where Φ is the transition matrix relating unit deviations in injection errors to deviations at time t. Then

$$PAR(t) = E(x(t)\; x(t)^T) = \Phi\, PAR(to)\, \Phi^T \qquad (1)$$

where PAR(t) is the covariance matrix of deviations from the nominal trajectory prior to any midcourse corrections.

Assuming we have a transition matrix which relates the vehicle state at time (t) to some desired conditions at the target, a guidance law may be derived to meet the desired end conditions. The two laws which the program can solve are: (1) Fixed Time of Arrival, and (2) Constant Energy with Respect to the Target.

$$\underline{C}(T) = (A_1 \;\; A_2) \begin{pmatrix} \underline{x}(t) \\ \dot{\underline{x}}(t) \end{pmatrix} \tag{2}$$

$$3 \times 1 \quad 3 \times 6 \quad 6 \times 1$$

where $\underline{C}(T)$ = target conditions at time T

$(A_1 \;\; A_2)$ = transition matrix

For Fixed Time of Arrival:

$$\underline{C}(T) = \begin{pmatrix} x(T) \\ y(T) \\ z(T) \end{pmatrix}$$

$$3 \times 1 \quad 3 \times 1$$

For Constant Energy with Respect to the Target:

$$\underline{C}(T) = \begin{pmatrix} \delta \; B \cdot T(T) \\ \delta \; B \cdot R(T) \\ \delta \; VINF \end{pmatrix}$$

The guidance law is derived such that the estimate of $\underline{C}(T)$, which represents deviations from the nominal trajectory at the target, is zero when the vehicle arrives at the target.

$$\underline{C}(T) = A_1 \; \hat{\underline{x}}(t) + A_2 \; \hat{\dot{\underline{x}}}(t)$$

$$3x1 \quad 3x3 \;\; 3x1 \quad 3x3 \;\; 3x1 \tag{3}$$

The velocity correction is selected such that:

$$\underline{C}(T) + A_2\, \underline{\dot{x}}_g = 0 = A_1\, \underline{\hat{x}}(t) + A_2\, \underline{\hat{\dot{x}}}(t) + A_2\, \underline{\dot{x}}_g \tag{4}$$

$\underline{\dot{x}}_g$ is multiplied by A_2 because as can be seen in equation (3), A_2 is a matrix of sensitivity partials relating $\underline{C}(T)$ to velocity deviations at time (t).

Solving equation 4 for $\underline{\dot{x}}_g$ yields

$$\underline{\dot{x}}_g = A_2^{-1}\, \underline{C}(T) = -A_2^{-1} A_1\, \underline{\hat{x}}(t) - I\, \underline{\hat{\dot{x}}}(t) \tag{5}$$

$$\underline{\dot{x}}_g = - \begin{bmatrix} A_2^{-1} A_1 & I \end{bmatrix} \begin{pmatrix} \underline{\hat{x}}(t) \\ \underline{\hat{\dot{x}}}(t) \end{pmatrix} \tag{6}$$

We are interested in the covariance matrix PAR at (T), that is, at the time of arrival, since this is the covariance matrix of errors in guidance.

If we find $E\,(x(t)\ x(t)^T)$ after a correction, the use of equation (1) will propagate it to the target.

Let subscript (a) define the state after a correction, and subscript (b) define the state before the correction.

Then

$$\begin{pmatrix} \underline{x}_a \\ \underline{\dot{x}}_a \end{pmatrix} = \begin{pmatrix} \underline{x}_b \\ \underline{\dot{x}}_b \end{pmatrix} + \begin{pmatrix} 0 \\ \underline{\dot{x}}_g \end{pmatrix} + \begin{pmatrix} 0 \\ q \end{pmatrix}$$

where $\dot{x}_g$ is the computed velocity correction and q is the error in making the correction. This is under the assumption of an impulsive correction. Utilizing equation (5), and since $x_a = \hat{x}_a + \tilde{x}_a$

$$\begin{pmatrix} \underline{x}_a \\ \dot{\underline{x}}_a \end{pmatrix} = \begin{pmatrix} I & 0 \\ -A_2A_1 & 0 \end{pmatrix} \begin{pmatrix} \hat{\underline{x}}_b \\ \hat{\dot{\underline{x}}}_b \end{pmatrix} + \begin{pmatrix} \tilde{\underline{x}}_b \\ \tilde{\dot{\underline{x}}}_b \end{pmatrix} + \begin{pmatrix} 0 \\ q \end{pmatrix} \tag{7}$$

and

$$\underset{6 \times 6}{E\,(x_a x_a^T)} = \underset{6 \times 6}{\begin{bmatrix} I & 0 \\ -A_2A_1 & 0 \end{bmatrix}} \underset{6 \times 6}{E(\hat{x}_b \hat{x}_b^T)} \underset{6 \times 6}{\begin{bmatrix} I & 0 \\ -A_2A_1 & 0 \end{bmatrix}^T} + \underset{6 \times 6}{E\,(\tilde{x}_b \tilde{x}_b^T)}$$

$$+ \underset{6 \times 6}{E\left[\begin{pmatrix} 0 \\ q \end{pmatrix} (0 \quad q)^T\right]} + \text{CROSS TERMS}$$

The cross terms are zero because:

1. $E\,(\hat{x}\,\tilde{x}^T) = 0$ because of using an optimum estimation procedure

2. $E\,(\hat{x}\, q_s^T) = E(\hat{x}\,\hat{x}^T)\, E(S) = 0$

   $E\,(\hat{x}\, q_p^T) = E(\hat{x}\,\hat{x}^T)\, E(U) = 0$

   for the two types of correction errors described below.

The two types of errors considered in making a guidance correction are shutoff error and pointing.

A. Shutoff Error

Let the correction by $\dot{\underline{x}}_g = \begin{pmatrix} V_x \\ V_y \\ V_z \end{pmatrix}$

The shutoff error is

$$\vec{\epsilon}_s = S\, \dot{\underline{x}}_g$$

where S is a scalar random variable which is gaussian (0, σ_s).

B. Pointing Error

The pointing error is $\vec{\epsilon}_p = \vec{U} \times \dot{\underline{x}}_g$

where $U = (U_x, U_y, U_z)$ is a three dimensional spherical gaussian random variable with sigmas equal to σ_p and mean 0.

The two types of expected values of interest in equation (8) for the error in correction are E (qq^T) and E $(\hat{x}\, q^T)$

$$\vec{q} = \vec{\epsilon}_p + \vec{\epsilon}_s$$

$$E(q\, q^T) = (\sigma_s^2 - \sigma_p^2)\, E(\dot{\underline{x}}_g\, \dot{x}_g^T) + \sigma_p^2\, E(\dot{x}_g^T\, \dot{x}_g)\, I$$

$$E(\hat{x}\, q^T) = 0$$

The two remaining expected values in equation (8) may be written as follows:

$$E(\hat{x}\, \hat{x}^T) = E(x - \tilde{x})\, (x - \tilde{x})^T = E(x\, x^T) - E(x\, \tilde{x}^T) + E(\tilde{x}\, \tilde{x}^T) - E(\tilde{x}\, x^T)$$

but $E(x\, \tilde{x}^T) = E\left([\hat{x} + \tilde{x}]\, \tilde{x}^T\right) = E\, (\tilde{x}\, \tilde{x}^T)$

since $E(\hat{x}\, \tilde{x}) = 0$

$$\therefore \quad E(\hat{x}\, \hat{x}^T) = E(x\, x^T) - E(\tilde{x}\, \tilde{x}^T)$$

$$E(\hat{x}\, \hat{x}^T) = PAR_b - P_b$$

Finally

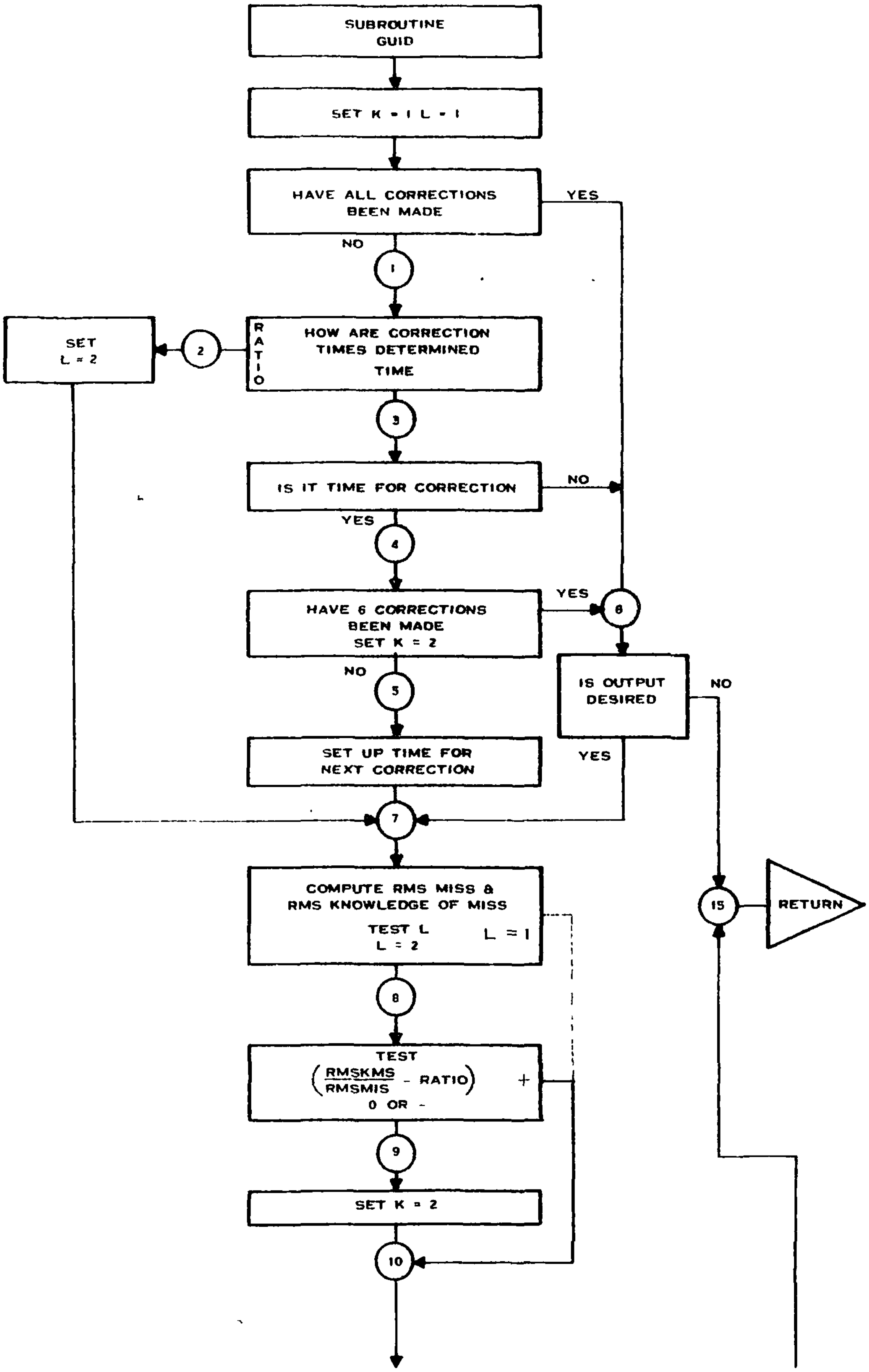
$$\underset{6\times 6}{PAR_a} = \underset{6\times 6}{\begin{bmatrix} I & 0 \\ -A_2^{1} A_1 & 0 \end{bmatrix}} \underset{6\times 6}{\begin{bmatrix} PAR_b - P_b \end{bmatrix}} \underset{6\times 6}{\begin{bmatrix} I & 0 \\ -A_2^{1} A_1 & 0 \end{bmatrix}^T}$$

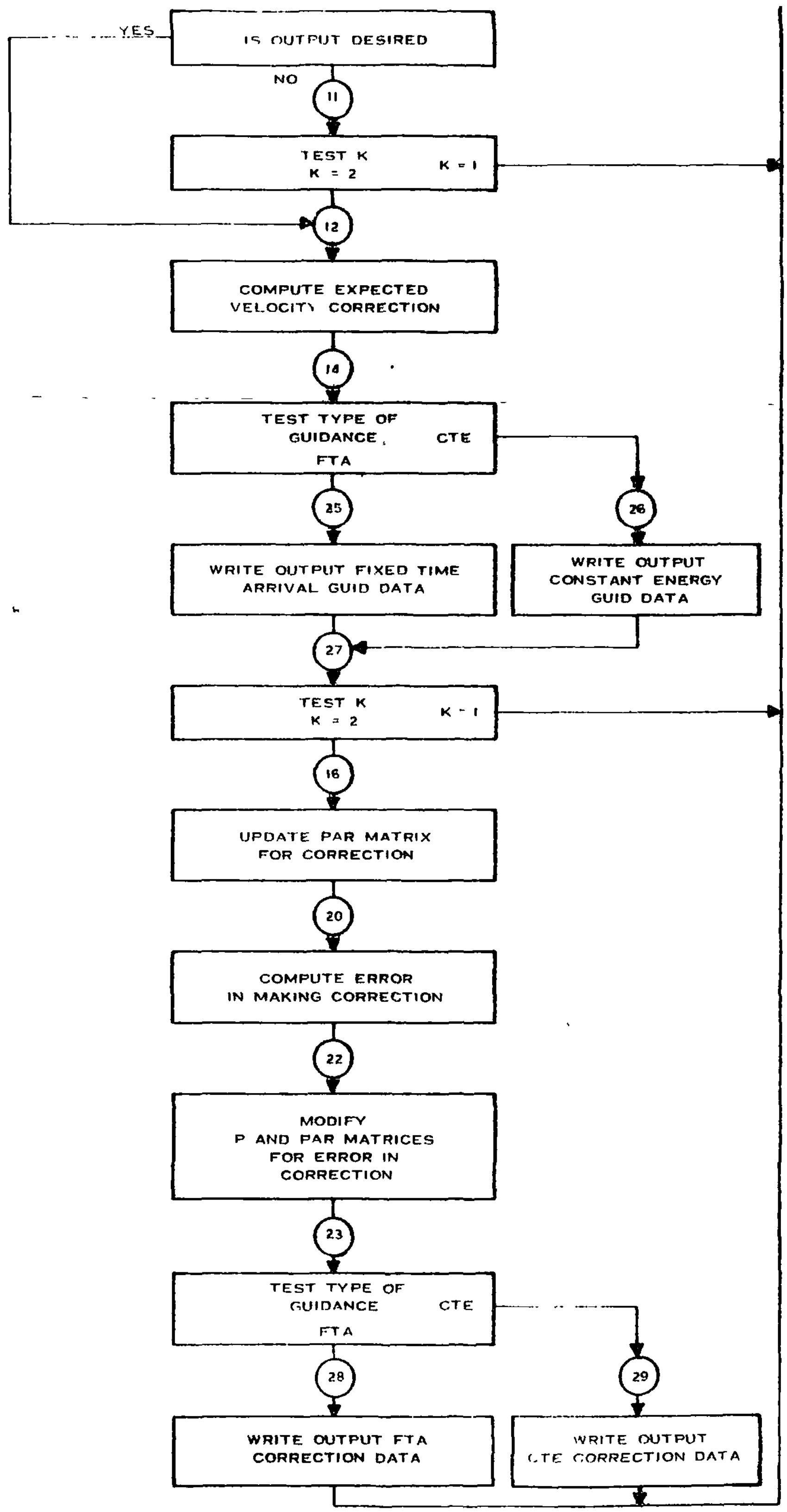
$$+ \underset{6\times 6}{P_b} + \underset{6\times 6}{\begin{bmatrix} 0 & 0 \\ 0 & E(qq^T) \end{bmatrix}}$$

The covariance matrix of the knowledge of the state is updated to account for the lack of exact knowledge of the correction.

$$\underset{6x6}{P_a} = \underset{6x6}{P_b} + \delta \underset{6x6}{\begin{bmatrix} 0 & 0 \\ 0 & E(qq^T) \end{bmatrix}}$$

where δ indicates the ability to monitor (ON BOARD) the error in correction.


SUBROUTINE GUID
SET K = 1 L = 1
HAVE ALL CORRECTIONS BEEN MADE
YES
NO
1
SET L = 2
2
RATIO
HOW ARE CORRECTION TIMES DETERMINED
TIME
3
IS IT TIME FOR CORRECTION
NO
YES
4
HAVE 6 CORRECTIONS BEEN MADE SET K = 2
YES
6
NO
5
IS OUTPUT DESIRED
NO
YES
SET UP TIME FOR NEXT CORRECTION
7
COMPUTE RMS MISS & RMS KNOWLEDGE OF MISS
TEST L
L = 2
L = 1
15
RETURN
8
TEST
(RMSKMS / RMSMIS - RATIO)
0 OR -
+
9
SET K = 2
10


IS OUTPUT DESIRED
YES
NO
11
TEST K
K = 2
K = 1
12
COMPUTE EXPECTED VELOCITY CORRECTION
14
TEST TYPE OF GUIDANCE
CTE
FTA
25
26
WRITE OUTPUT FIXED TIME ARRIVAL GUID DATA
WRITE OUTPUT CONSTANT ENERGY GUID DATA
27
TEST K
K = 2
K = 1
16
UPDATE PAR MATRIX FOR CORRECTION
20
COMPUTE ERROR IN MAKING CORRECTION
22
MODIFY P AND PAR MATRICES FOR ERROR IN CORRECTION
23
TEST TYPE OF GUIDANCE
CTE
FTA
28
29
WRITE OUTPUT FTA CORRECTION DATA
WRITE OUTPUT CTE CORRECTION DATA

```
*      LABEL                                                           GUID
*      SYMBOL TABLE                                                    GUID
CEC2005  SUBROUTINE GUID                                               GUID
       SUBROUTINE GUID                                                 GUID0000
       COMMON T,S,C,IC                                                 GUID0010
       DIMENSION T(1360),S(1000),C(1000),IC(1)                         GUID0020
      1,TPAR(3,6),PAR(6,6),P(6,6),DUP(3,3)                             GUID0030
      2,TGUID(6),DUM(3,3),DUN(3,3),DUMM(6,6)                           GUID0040
       EQUIVALENCE (T,TDUM),(S,SDUM),(C,CDUM)                          GUID0050
      1,(IC,ICDUM),(C(568),TPAR),(C(460),PAR)                          GUID0060
      2,(C(652),P),(IC(218),N),(C(648),TGUIDE)                         GUID0070
      3,(IC(219),IGD),(IC(217),IGDTP),(IC(214),NOUT)                   GUID0080
      4,(C(30),TSEC),(S(476),TGUID),(S(483),RATIO)                     GUID0090
      5,(S(473),QQPT),(S(471),QQSO),(S(484),DMONIT)                    GUID0100
       CALL SETN(NIN,NUTS)                                             GUID0105
C      IGD=NUMBER OF GUIDANCE CORRECTIONS TO BE MADE                   GUID0110
C      N KEY USED TO TEST NUMBER OF GUIDANCE CORRECTIONS WHICH HAVE    GUID0120
C      BEEN MADE                                                       GUID0130
C      S(475)  0=GUIDE ON INPUT TIMES   + GUIDE ON A RATIO TEST        GUID0140
C      IGDTP 1.FTA  2.CTE  3.MINIMUM FUEL                              GUID0150
C      DMONIT=ACCURACY OF MONITORING THE GUIDANCE CORRECTION           GUID0160
       NOUT=NOUT                                                       GUID0170
       K=1                                                             GUID0180
       L=1                                                             GUID0190
       IF(IGD-N)6,1,1                                                  GUID0200
 1     IF(S(475))3,3,2                                                 GUID0210
  2    L=2                                                             GUID0220
       GO TO 7                                                         GUID0230
 3     IF(TGUIDE-TSEC)4,4,6                                            GUID0240
 4     CONTINUE                                                        GUID0250
       K=2                                                             GUID0260
       N=N+1                                                           GUID0270
       IF(N-7)5,6,6                                                    GUID0280
     5 CALL TIMED(TGUID(N),TGUIDE)                                     GUID0290
       GO TO 7                                                         GUID0300
 6     CONTINUE                                                        GUID0310
       GO TO (15,7),NOUT                                               GUID0320
 7     CALL MATRX(TPAR,P,DUM,3,6,6,1)                                  GUID0330
       RMSKMS=SQRTF(DUM(1,1)+DUM(2,2)+DUM(3,3))                        GUID0340
       CALL MATRX(TPAR,PAR,DUN,3,6,6,1)                                GUID0350
       RMSMIS=SQRTF(DUN(1,1)+DUN(2,2)+DUN(3,3))                        GUID0360
       GO TO (10,8),L                                                  GUID0370
 8     IF((RMSKMS/RMSMIS)-RATIO)9,9,10                                 GUID0380
 9     K=2                                                             GUID0390
       N=N+1                                                           GUID0400
 10    GO TO (11,12),NOUT                                              GUID0410
 11    GO TO (15,12),K                                                 GUID0420
 12    CONTINUE                                                        GUID0430
       DO 13 I=1,3                                                     GUID0440
       DO 13 J=1,3                                                     GUID0450
       JJ=J+3                                                          GUID0460
```

GUID (Cont'd)

```
      DUP(I,J)=-TPAR(I,JJ)                                               GUID047
   13 CONTINUE                                                           GUID048
      CALL INV3(DUP,3,DET)                                               GUID049
      DO 14 I=1,3                                                        GUID050
      DO 14 J=1,3                                                        GUID051
      DUM(I,J)=DUN(I,J)-DUM(I,J)                                         GUID052
   14 CONTINUE                                                           GUID053
      RMSPAP=SQRTF(PAR(1,1)+PAR(2,2)+PAR(3,3))                           GUID054
      RMSPAV=SQRTF(PAR(4,4)+PAR(5,5)+PAR(6,6))                           GUID055
      CALL MATRX(DUP,DUM,DUM,3,3,3,1)                                    GUID056
C     DUM IS THE COVARIANCE MATRIX OF INDICATED VELOCITY CORRECTION      GUID057
      RMSV2=DUM(1,1)+DUM(2,2)+DUM(3,3)                                   GUID058
      RMSV=SQRTF(RMSV2)                                                  GUID059
      IGDTP=IGDTP                                                        GUID059
      GO TO (25,26),IGDTP                                                GUID059
   25 CONTINUE                                                           GUID059
      WRITE OUTPUT TAPE NUTS,800,RMSMIS,RMSPAP,RMSKMS,RMSPAV,RMSV        GUID060
  800 FORMAT(18H0GUID DATA FOLLOWS,/                                     GUID061
     124H    RMS FTA TARGET MISS=,E15.8,                                 GUID062
     225H    RMS POS DEV FROM NOM=,E15.8,/                               GUID063
     324H      RMS KNOW. OF MISS=,E15.8,                                 GUID064
     425H    RMS VEL DEV FROM NOM=,E15.8,/                               GUID065
     524H             RMS VEL REQ=,E15.8)                                GUID065
      GO TO 27                                                           GUID065
   26 CONTINUE                                                           GUID065
      RMSMIS=SQRTF(DUN(1,1)+DUN(2,2))                                    GUID065
      RMSVIN=SQRTF(DUN(3,3))                                             GUID065
      WRITE OUTPUT TAPE NUTS,802,RMSMIS,RMSPAP,RMSVIN,RMSPAV,RMSKMS,RMSVGUID066
  802 FORMAT(18H0GUID DATA FOLLOWS,/                                     GUID066
     124H    RMS TARGET POS MISS=,E15.8,                                 GUID066
     225H    RMS POS DEV FROM NOM=,E15.8,/                               GUID066
     324H     RMS VINFINITY MISS=,E15.8,                                 GUID066
     425H    RMS VEL DEV FROM NOM=,E15.8,/                               GUID066
     324H      RMS KNOW. OF MISS=,E15.8,                                 GUID066
     525H             RMS VEL REQ=,E15.8)                                GUID066
   27 CONTINUE                                                           GUID066
      GO TO (15,16),K                                                    GUID067
   15 RETURN                                                             GUID068
   16 CONTINUE                                                           GUID069
C     FOLLOWING IS UPDATING OF PAR MATRIX FOR GUIDANCE CORRECTION        GUID070
      DO 17 I=1,6                                                        GUID071
      DO 17 J=I,6                                                        GUID072
      PAR(I,J)=PAR(I,J)-P(I,J)                                           GUID073
      PAR(J,I)=PAR(I,J)                                                  GUID074
   17 CONTINUE                                                           GUID075
      CALL MATRX(DUP,TPAR,DUP,3,3,3,0)                                   GUID076
      DO 19 I=1,3                                                        GUID077
      II=I+3                                                             GUID078
      DO 18 J=1,3                                                        GUID079
      JJ=J+3                                                             GUID080
      DUMM(II,J)=DUP(I,J)                                                GUID081
```

GUID (Cont'd)

```
      DUMM(I,JJ)=0.                                                     GUID0820
      DUMM(II,JJ)=0.                                                    GUID0830
   18 DUMM(I,J)=0.                                                      GUID0840
   19 DUMM(I,I)=1.                                                      GUID0850
      CALL MATRX(DUMM,PAR,PAR,6,6,6,1)                                  GUID0860
      DO 20I=1,6                                                        GUID0870
      DO 20 J=I,6                                                       GUID0880
      PAR(I,J)=PAR(I,J)+P(I,J)                                          GUID0890
      PAR(J,I)=PAR(I,J)                                                 GUID0900
 20   CONTINUE                                                          GUID0910
      CRUD=QQPT*RMSV2                                                   GUID0920
      QQ=QQSO-QQPT                                                      GUID0930
      DO 22 I=1,3                                                       GUID0940
      DO 21 J=I,3                                                       GUID0950
      DUN(I,J)=QQ*DUM(I,J)                                              GUID0960
      DUN(J,I)=DUN(I,J)                                                 GUID0965
 21   CONTINUE                                                          GUID0970
      DUN(I,I)=DUN(I,I)+CRUD                                            GUID0980
 22   CONTINUE                                                          GUID0990
      RMSERR=SQRTF(DUN(1,1)+DUN(2,2)+DUN(3,3))                          GUID1000
      DO 23 I=1,3                                                       GUID1010
      II=I+3                                                            GUID1020
      DO 23 J=I,3                                                       GUID1030
      JJ=J+3                                                            GUID1040
      PAR(II,JJ)=PAR(II,JJ)+DUN(I,J)                                    GUID1050
      PAR(JJ,II)=PAR(II,JJ)                                             GUID1060
      P(II,JJ)=P(II,JJ)+DMONIT*DUN(I,J)                                 GUID1070
      P(JJ,II)=P(II,JJ)                                                 GUID1080
   23 CONTINUE                                                          GUID1090
      CALL MATRX(TPAR,PAR,DUN,3,6,6,1)                                  GUID1100
      RMSPAP=SQRTF(PAR(1,1)+PAR(2,2)+PAR(3,3))                          GUID1105
      RMSPAP=SQRTF(PAR(1,1)+PAR(2,2)+PAR(3,3))                          GUID1110
      RMSPAV=SQRTF(PAR(4,4)+PAR(5,5)+PAR(6,6))                          GUID1115
      GO TO (28,29),IGDTP                                               GUID1120
   28 CONTINUE                                                          GUID1125
      RMSMIS=SQRTF(DUN(1,1)+DUN(2,2)+DUN(3,3))                          GUID1130
      WRITE OUTPUT TAPE NUTS,801,RMSERR,RMSPAP,RMSMIS,RMSPAV            GUID1140
 801  FORMAT(32H  ***GUIDANCE CORRECTION MADE***,                        GUID1150
     1/24H     RMS ERROR IN CORR.=,E15.8,                                GUID1160
     225H    RMS POS DEV FROM NOM=,E15.8,                                GUID1170
     3/24H      RMS MIS AFT CORR.=,E15.8,                                GUID1180
     425H    RMS VEL DEV FROM NOM=,E15.8)                                GUID1190
      GO TO 15                                                          GUID1200
   29 CONTINUE                                                          GUID1210
      RMSMIS=SQRTF(DUN(1,1)+DUN(2,2))                                   GUID1220
      RMSVIN=SQRTF(DUN(3,3))                                            GUID1230
      WRITE OUTPUT TAPE NUTS,803,RMSERR,RMSPAP,RMSMIS,RMSPAV,RMSVIN     GUID1240
 803  FORMAT(32H  ***GUIDANCE CORRECTION MADE***,                        GUID1250
     1/24H     RMS ERROR IN CORR.=,E15.8,                                GUID1260
     225H    RMS POS DEV FROM NOM=,E15.8,/                               GUID1270
     3/24H  RMS POS MIS AFT CORR.=,E15.8,                                GUID1280
```

```
    425H    RMS VEL DEV FROM NOM=,E15.8,                                          GUID129
    5/24H     RMS VINF AFT CORR.=,E15.8)                                          GUID130
 30  CONTINUE                                                                     GUID131
     GO TO 15                                                                     GUID132
     END                                                                          GUID
```

Page Intentionally Left Blank

Subroutine: HOUR

Purpose: To provide a reading of the printer clock in hours since midnight. This is the STL subroutine RW HOUR.

Calling Sequence:

CALL HOUR(A)

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
O	A			hours	Time

Common storages used or required: None

Subroutines required: None

Functions required: None

Approximate number of storages required: 86 cells

Restrictions

Clock advance is suppressed during readout so that elapsed time will be off .4N seconds where N is the number of times the clock is selected.

HOUR will always provide a floating point number if the printer is available. Therefore, a faulty clock or the wrong board will be indicated.

Use

The main application of HOUR is to time cases through use of the FINP "CLOCK" option.

HOUR

```
*       FAP
*              HOUR SUBROUTINE 10-27-31
        ENTRY  HOUR
HOUR    CLA    1,4
        STA    SSAM
        SXA    SAMX,4
        SXA    SAMX+1,2
        TCOA   *                DELAY

        RPRA                    ECHO CHECK
        SPRA   6                READ CLOCK
        SPRA   9                SUPPRESS SPACING
        SPRA   10               AND PRINTING
        AXT    24,4
        STZ    SAMZ+24,4
        TIX    *-1,4,1
        RCHA   SAMIO            TRANSMIT CLOCK READING
        TCOA   *
        STZ    SAMZ+18
        STZ    SAMZ+20
        AXT    18,4
SAM1    CAL    SAMZ+18,4        8-4 LEFT ADD ECHOS
        ORS    SAMZ+18          AND CONVERT TO
        LDQ    SAMZ+18          BCD
        PXD
        AXT    5,2
SAM3    LGL    1
        ALS    5
        TIX    SAM3,2,1
        LGL    1
        ACL    SAMZ+20
        SLW    SAMZ+20
        TIX    SAM1,4,2
        STZ    SAMZ+22
        AXT    6,4
        LDQ    SAMZ+20
SAM4    PXD
        LGL    6
        SLW    SAMZ+23
        CLA    SAMZ+22
        ALS    2
        ADD    SAMZ+22
        ALS    1
        ADD    SAMZ+23
        STO    SAMZ+22
        TIX    SAM4,4,1
        LDQ    SAMZZ
        DVP    SIX
        XCL
        ORA    SA
        FAD    SA
```

HOUR (Cont'd)

```
         CAS      M24                                           HOUR0500
         FSB      M24                 MOD 24 HOURS              HOUR0510
         NOP                          MIDNIGHT ONLY             HOUR0520
SSAM     STO      **0                                           HOUR0530
SAMX     AXT      **0,4                                         HOUR0540
         AXT      **0,2                                         HOUR0550
         TRA      2,4                 EXIT                      HOUR0560
SAMIO    IOCP     SAMZ,2+0,26         NON TRANSMIT 26 COPIES    HOUR0570
         IOCP     SAMZ,0,1            27TH COPY 9L ECHO         HOUR0580
         IOCP     SAMZ+1,2,3          NON TRANSMIT 9R,12L,12R   HOUR0590
         IOCD     SAMZ+2,0,16         8L TO 1R ECHOS            HOUR0600
SA       OCT      214000000000                                  HOUR0610
M24      DEC      24,                                           HOUR0620
SIX      DEC      6000B15                                       HOUR0630
SAMZZ    PZE                                                    HOUR0640
SAMZ     BSS      24                                            HOUR0650
         END                                                    HOUR0660
```

Subroutine: HPHT

Purpose: To perform the matrix multiplication $(RMS)^2 = H \cdot P \cdot H^T$. The square root of $(RMS)^2$ is taken and RMS, which is the standard deviation of the parameter being considered, is output from the subroutine.

Calling Sequence:

CALL HPHT (H, P, RMS)

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	H	(1,6)	$\frac{\partial \text{ PAR}}{\partial \text{ STATE}}$		Parameter Partial Matrix
I	P	(6,6)	$E(\vec{X}\,\vec{X}^T)$		State Covariance Matrix
O	RMS	1			Standard deviation of parameter
					due to state variance.

Common storages used or required: None

Subroutines required: None

Functions required: SQRT

Approximate number of storages required: ______

HPHT

```
*     LABEL                                                          HPHT
*     SYMBOL TABLE                                                   HPHT
      SUBROUTINE HPHT(H,P,RMS)                                       HPHT0000
      DIMENSION H(6),P(6,6),DUM(6)                                   HPHT0010
      DO 1 I=1,6                                                     HPHT0020
      DUM(I)=0.                                                      HPHT0030
      DO 1 J=1,6                                                     HPHT0040
      DUM(I)=DUM(I)+H(J)*P(J,I)                                      HPHT0050
    1 CONTINUE                                                       HPHT0060
      VAR2=0.                                                        HPHT0070
      DO 2 I=1,6                                                     HPHT0080
      VAR2=VAR2+DUM(I)*H(I)                                          HPHT0090
    2 CONTINUE                                                       HPHT0100
      RMS=SQRTF(VAR2)                                                HPHT0110
      RETURN                                                         HPHT0120
      END                                                            HPHT
```

Subroutine: INPUT

Purpose: To transform input initial conditions of position and velocity in various coordinate systems to Cartesian coordinates with respect to earth's equator and equinox of 1950. Input may be Cartesian referred to (1) mean equator and equinox, 1950; (2) true equator and equinox of date; (3) earth-fixed or spherical; (4) mean equator and equinox of 1950; (5) true equator and equinox of date; (6) earth-fixed.

Calling Sequence:

CALL INPUT (K, XOUT, VOUT, TW, TF, AN, XIN, VIN)

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	K	1			Type of input: K=1 for (1) above, etc.
O	XOUT	3		see following pgs	Position, equinox 1950.0
O	VOUT	3			Velocity, equinox 1950.0
I	TW	1		days	Whole number of days past 1950.
I	TF	1		days	Fractional number of days past TW
O	AN	3,3			Transformation matrix from system
					(1) to (2)
I	XIN	3		see following pgs	Input position vector
I	VIN	3		see following pgs	Input velocity vector

Common storages used or required: None

Subroutines required: ROTEQ, NUTAIT, MULT, RVIN, INV3, GHA, SETN

Functions required: SIN, COS

Approximate number of storages required:

Input format:

For Cartesian coordinates: [K = 1,2,3]

XIN(1) = X
XIN(2) = Y
XIN(3) = Z
VIN(1) = $\dot{X}$
VIN(2) = $\dot{Y}$
VIN(3) = $\dot{Z}$

For sperical coordinates: [K = 4,5,6]

XIN(1) = R = magnitude of radius vector
XIN(2) = O = declination (degrees)
XIN(3) = θ = right ascension (degrees)
VIN(1) = V = magnitude of velocity
VIN(2) = γ = flight path angle (degrees)
VIN(3) = σ = azimuth angle (degrees)

Input Transformations:

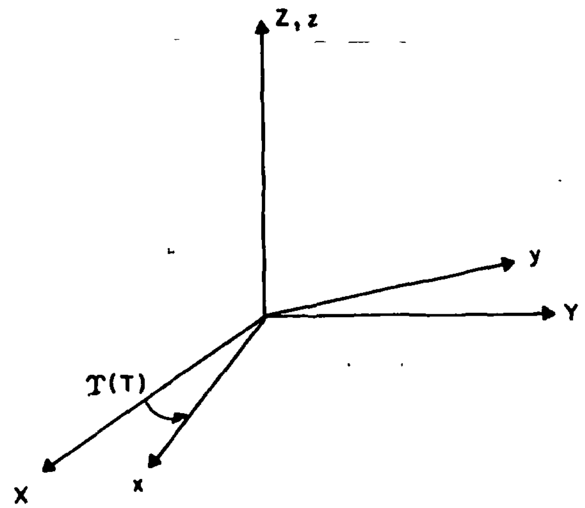
K = 1 The input is in mean equinox of 1950 and no transformation is required

$$\underset{3\times1}{\vec{XOUT}_{50}} = \underset{3\times1}{\vec{XIN}_{50}} \qquad \underset{3\times1}{\vec{VOUT}_{50}} = \underset{3\times1}{\vec{VIN}_{50}}$$

K = 2 Subroutines ROTEQ and NUTAIT are called to obtain the transformation matrix, AN, from mean equator 1950 to true equator of date

$$\underset{3\times1}{\vec{XOUT}_{50}} = \underset{3\times3}{(AN)^T} \underset{3\times1}{XIN_{DATE}} \qquad \underset{3\times1}{\vec{VOUT}_{50}} = \underset{3\times3}{(AN)^T} \underset{3\times1}{\vec{VIN}_{DATE}}$$

K = 3 The earth fixed Cantesian coordinate system is assumed to rotate with the earth. The X-Y plane is coincident with the earth's true equator of date, the X axis lying in the Greenwich meridian and the Z axis along the earth's spin axis. Subroutine GHA is called with time, T, to obtain the Greenwich hour angle, $\gamma^{(T)}$, of the true vernal equinox of date. The transformations from earth fixed coordinates to true equator of date are the following:



$$\underset{3\times1}{X_{DATE}} = \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}_{DATE} = \begin{pmatrix} \cos^{\gamma} & -\sin^{\gamma} & 0 \\ \sin^{\gamma} & \cos^{\gamma} & \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} z \\ y \\ z \end{pmatrix}_{EF}$$

and

$$V_{DATE} = \begin{pmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{pmatrix}_{DATE} = \begin{pmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix}_{EF} +$$

$$\underset{3x1}{}$$

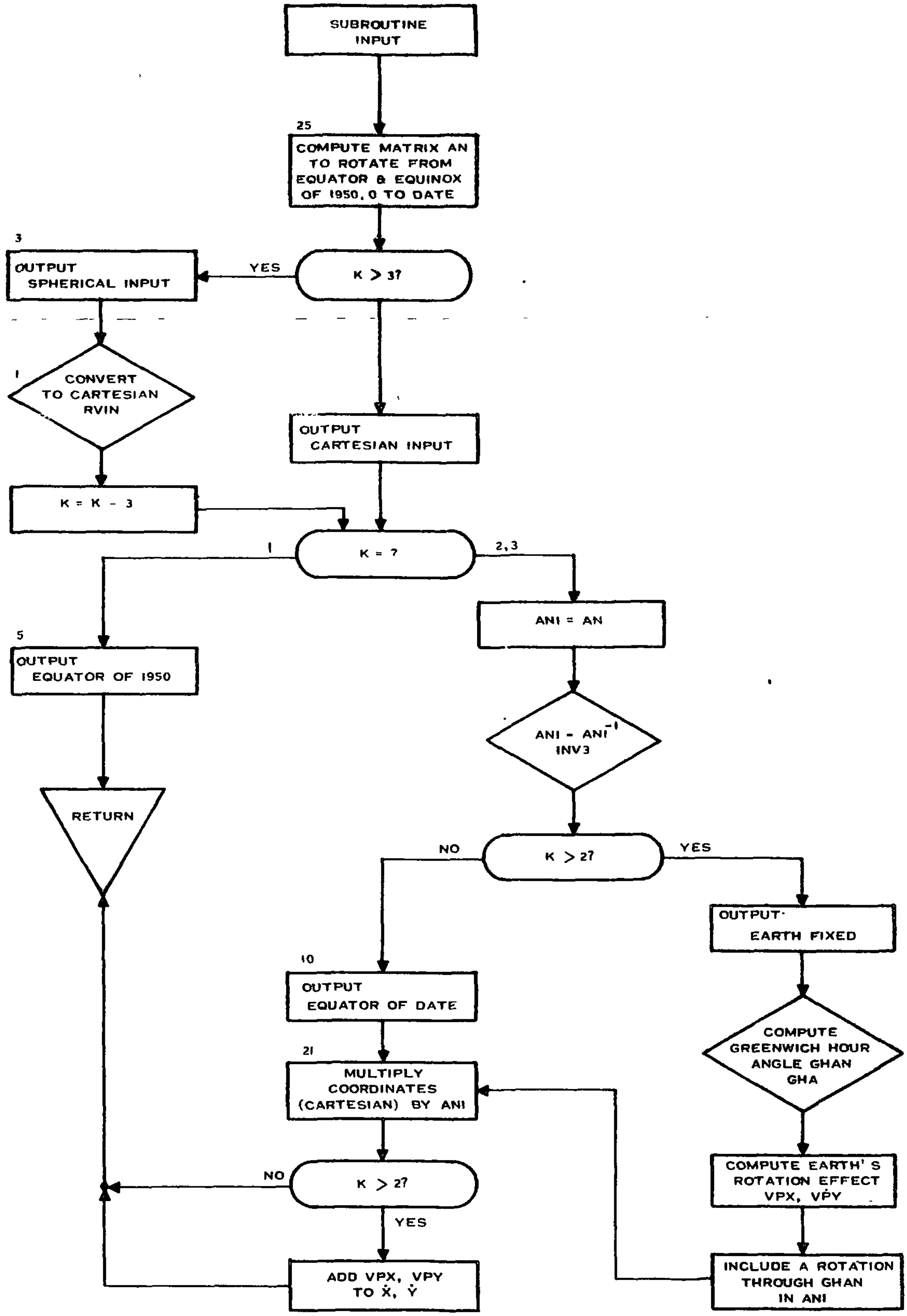
$$\omega \begin{pmatrix} -\sin\gamma & -\cos\gamma & 0 \\ \cos\gamma & -\sin\gamma & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix}_{EF}$$

where $\omega = \frac{d}{dt}\{\gamma(t)\}$ = earth's rotation rate

Subroutines ROTEQ and NUTAIT are called to obtain the AN transformation which completes the transformation to equator 1950.

$$\underset{3x1}{\vec{X}OUT} = \underset{3x3}{(AN)^T} \underset{3x1}{\vec{X}_{DATE}} \qquad \underset{3x1}{\vec{V}OUT} = \underset{3x3}{(AN)^T} \underset{3x1}{\vec{V}_{DATE}}$$

<u>K = 4, 5, or 6</u> In each of these cases, subroutine RVIN is called to transform the spherical sets of coordinates to Cartesian coordinates of the same reference frame. Following this spherical to Cartesian conversion, the transformation procedures which are followed for the transformation to 1950 are identical to those described for the corresponding reference frames for K = 1, 2, or 3.


SUBROUTINE INPUT
25
COMPUTE MATRIX AN TO ROTATE FROM EQUATOR & EQUINOX OF 1950, 0 TO DATE
K > 3?
YES
3
OUTPUT SPHERICAL INPUT
1
CONVERT TO CARTESIAN RVIN
K = K - 3
OUTPUT CARTESIAN INPUT
K = ?
1
2,3
5
OUTPUT EQUATOR OF 1950
RETURN
ANI = AN
ANI - ANI⁻¹ INV3
K > 2?
NO
YES
OUTPUT EARTH FIXED
10
OUTPUT EQUATOR OF DATE
21
MULTIPLY COORDINATES (CARTESIAN) BY ANI
COMPUTE GREENWICH HOUR ANGLE GHAN GHA
COMPUTE EARTH'S ROTATION EFFECT VPX, VṖY
K > 2?
NO
YES
ADD VPX, VPY TO Ẋ, Ẏ
INCLUDE A ROTATION THROUGH GHAN IN ANI

INPUT

```
*      LABEL                                                                INPT
*      SYMBOL TABLE                                                         INPT
CEC2013  SUBROUTINE INPUT                                                   INPT
       SUBROUTINE INPUT(K,X,V,TW,TF,AN,XIN,VIN)                             INPT0000
C      XIN AND VIN ARE INPUTS, X AND V ARE OUTPUTS IN EQUINOX 1950          INPT0010
C      K IS TYPE INPUT (1) EQUATOR 1950, (2) EQUATOR OF DATE (3)            INPT0015
C      EARTH-FIXED                                                          INPT0016
C      (4),(5),(6), ARE SPHERICAL INPUT RESP,,(1),(2),(3)                   INPT0020
       DIMENSION X(3),V(3),SP(6),EN(3,3),A(3,3),AN(3,3),ANI(3,3),B(3,3)     INPT0030
       DIMENSION XIN(3),VIN(3)                                              INPT0040
       CALL SETN(NIN,NOUT)                                                  INPT0045
       TD = TW+TF                                                           INPT0050
       DO 25 I=1,3                                                          INPT0060
       X(I)=XIN(I)                                                          INPT0070
       V(I)=VIN(I)                                                          INPT0080
   25  CONTINUE                                                             INPT0090
       CALL ROTEQ(TD,A)                                                     INPT0100
       CALL NUTAIT (TD,OM,CRUD,DDC,EN,EPSIL)                                INPT0110
       CALL MULT(EN,A,AN,0)                                                 INPT0120
       IF (K-3) 4,4,3                                                       INPT0130
    4  WRITE OUTPUT TAPE NOUT,15,X,V                                        INPT0140
   15  FORMAT(17H CARTESIAN INPUT 6E17.8)                                   INPT0150
       GO TO 16                                                             INPT0160
    3  WRITE OUTPUT TAPE NOUT,17,X,V                                        INPT0170
   17  FORMAT(17H SPHERICAL INPUT 6E17.8)                                   INPT0180
       DO 2 I=1,3                                                           INPT0190
       J = I+3                                                              INPT0200
       SP(I) = X(I)                                                         INPT0210
    2  SP(J) = V(I)                                                         INPT0220
    1  CALL RVIN(SP,X,V)                                                    INPT0230
       K = K-3                                                              INPT0240
   16  DO 12 I=1,3                                                          INPT0250
       J = I+3                                                              INPT0260
       SP(I) = X(I)                                                         INPT0270
   12  SP(J) = V(I)                                                         INPT0280
       GO TO (5,7,7), K                                                     INPT0290
    5  WRITE OUTPUT TAPE NOUT,18                                            INPT0300
   18  FORMAT(16H EQUATOR OF 1950)                                          INPT0310
       GO TO 13                                                             INPT0320
    7  CONTINUE                                                             INPT0330
       DO 6 I=1,3                                                           INPT0340
       DO 6 J=1,3                                                           INPT0350
    6  ANI(I,J) = AN(I,J)                                                   INPT0360
       CALL INV3(ANI,3,D)                                                   INPT0370
       IF (K-2) 10,10,8                                                     INPT0380
    8  DD = EN(2,1)                                                         INPT0390
       WRITE OUTPUT TAPE NOUT,20                                            INPT0400
   20  FORMAT(12H EARTH FIXED)                                              INPT0410
       TS = TF*86400.                                                       INPT0420
       CALL GHA(TS,TW,GHAN,DD,OMEGA)                                        INPT0430
       OMEGA = OMEGA*.017453296                                             INPT0440
```

INPUT (Cont'd

```
      GHAN = GHAN*.017453296                                    INPT04
      SP(4)=SP(4)-OMEGA*SP(2)                                   INPT04
      SP(5)=SP(5)+OMEGA*SP(1)                                   INPT04
      CALL ROTATE(3,GHAN,B,-1)                                  INPT04
      CALL MULT(ANI,B,EN,-1)                                    INPT04
      DO 9 I=1,3                                                INPT05
      DO 9 J=1,3                                                INPT05
    9 ANI(I,J) = EN(I,J)                                        INPT05
      GO TO 21                                                  INPT05
   10 WRITE OUTPUT TAPE NOUT,19                                 INPT05
   19 FORMAT(16H EQUATOR OF DATE)                               INPT05
   21 DO 11 I=1,3                                               INPT05
      X(I) = 0.                                                 INPT05
      V(I) = 0.                                                 INPT05
      DO 11 J=1,3                                               INPT05
      X(I) = X(I) + ANI(I,J)*SP(J)                              INPT06
   11 V(I) = V(I) + ANI(I,J)*SP(J+3)                            INPT06
   13 RETURN                                                    INPT06
      END                                                       INPT
```

Page Intentionally Left Blank

Subroutine: INTR, INTRI

Purpose: The purpose of INTR is to interpolate as a function of the central body on the coordinates of the other bodies, to the given time, and return the interpolated position values. The purpose of INTRI is the same as INTR, but in addition it numerically differentiates the positions to obtain the velocities and also return them.

Calling Sequence:

CALL INTR, INTRI (TW, TF, NB, PO, NV, VE, DIS)

Input and **Output**

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	TW	1		whole days	Time in days since 1950.0
I	TF	1		fractional days	Time in fractional days since TW
I	NB	1			Central body
O	PO	22		Km	Array of position values
I	NV	1			Dummy variable
O	VE	22		Km/sec	Array of velocity values
I	DIS	1		Km	Jupiter included if DIS $> 10^6$

Common storages used or required: C

Subroutines required: FIX, FLOAT

Functions required:

Approximate number of storages required: 2063 OCTAL

For interpolation, the following formula is used:

$$y(t) = \left\{ uy_o + ty_1 \right\} + \left\{ \frac{u(u^2-1)}{3!} \delta^2 y_o + \frac{t(t^2-1)}{3!} \delta^2 y_1 \right\}$$

$$+ \left\{ \frac{u(u^2-1)(u^2-4)}{5!} \delta^4 y_o + \frac{t(t^2-1)(t^2-4)}{5!} \delta^4 y_1 \right\}$$

where $y_o = y(T_j)$

$y_1 = y(T_j + h)$

h = ephemeris interval

$t = \frac{T-T_j}{h}$

$u = 1-t$

$T_j \leq T < T + h$

For the velocity values, the following is used:

$$\dot{y}(T) = \frac{1}{h} \left\{ -y_o + y_1 \right\} + \frac{1}{h} \left\{ - \frac{3u^2-1}{3!} \delta^2 y_o + \frac{3t^2-1}{3!} \delta^2 y_1 \right\}$$

$$+ \frac{1}{h} \left\{ - \frac{5u^4-15u^2+4}{5!} \delta^4 y_o + \frac{5t^4-15t^2+4}{5!} \delta^4 y_1 \right\}$$

The locations of variables in the position and velocity arrays are as follows:

PO(5)	Z value of Jupiter
PO(6)	Y value of Jupiter
PO(7)	X value of Jupiter

PO(8) - PO(10)	values of Mars
PO(11) - PO(13)	values of Venus
PO(14) - PO(16)	values of the Sun
PO(17) - PO(19)	values of the Moon
PO(20) - PO(22)	values of the Earth

The VE array is arranged the same as the PO array with velocity values in place of positions.

```
      FAP                                                              INTR
       TTL JPL EPHEMERIS LOOKUP   NEW ERP 6/63                         INTR0000
       LBL INTR                                                        INTR0010
       COUNT   530                                                     INTR0020
       ENTRY   INTR                                                    INTR0030
       ENTRY   INTR1                                                   INTR0040
       REM     CALL INTR A,B,C,D,E,F,G  POSITIONS AND VELOCITIES      INTR0050
       REM     A=ARG INTEGRAL DAYS SINCE 1950.0                        INTR0060
       REM     B=FRACTION OF DAY                                       INTR0070
       REM     C=CENTRAL BODY 0=EARTH 1=MOON 2=SUN 3=VEN 4=MARS        INTR0080
       REM     D=OUTPUT TABLE STARTING ADDRESS                         INTR0090
       REM     E=IGNORED (VELOCITY OPTION INDICATOR)                   INTR0100
       REM     F=ADDRESS OF VELOCITY TABLE                             INTR0110
       REM     G=RADIUS FROM CENTRAL BODY                              INTR0120
       REM     EPHEMERIS TAPE MUST BE MOUNTED ON A6                    INTR0130
INTR   TXL     *+3,,**                                                 INTR0140
INTR1  SXD     VEL,4                                                   INTR0150
       TXL     *+2,,**                                                 INTR0160
       STZ     VEL                                                     INTR0170
       CLA     1,4                                                     INTR0180
       STA     TAR                                                     INTR0190
       CLA     2,4                      FRA                            INTR0200
       STA     FRA                                                     INTR0210
       CLA     3,4                      CEN                            INTR0220
       STA     CEN                                                     INTR0230
       CLA     4,4                      OUP                            INTR0240
       STA     OUP                                                     INTR0250
       CLA     6,4                      VELOCITY                       INTR0260
       STA     XN,V                                                    INTR0270
       CLA     7,4                                                     INTR0280
       STA     RADS                                                    INTR0290
TAR    CLA     **                                                      INTR0300
       STO     TARG                                                    INTR0310
FRA    CLA     **                                                      INTR0320
       STO     TARG+1                                                  INTR0330
CEN    CLA     **                                                      INTR0340
       ARS     18                                                      INTR0350
       STO     CENTER                                                  INTR0360
RADS   CLA     **                                                      INTR0370
       STO     R                                                       INTR0380
       SXD     TRAP,4                                                  INTR0390
       SXD     TRAP+1,2                                                INTR0400
       SXD     HELIO-1,1                                               INTR0410
       CLA     VEL                                                     INTR0420
       TNZ     NEU                      MUST INTERPOLATE TO            INTR0430
       REM                              OBTAIN VELOCITY                INTR0440
       CLA     CENTER                                                  INTR0450
       SUB     KERNO                                                   INTR0460
       TNZ     NEU                      MUST INTERPOLATE FOR           INTR0470
       REM                              NEW CENTRAL BODY               INTR0480
       CLA     TARG                                                    INTR0490
```

INTR, INTRI(Cont'd)

```
         SUB     TARGO                                        INTR050
         TNZ     NEU                TIME HAS CHANGED          INTR051
         CLA     TARG+1                                       INTR052
         SUB     TARGO+1                                      INTR053
         TZE     FLEE+6                                       INTR054
         REM                        ANALYZE TABLE             INTR055
         REM                        NEEDS AS A                INTR056
         REM                        FUNCTION OF               INTR057
         REM                        CENTRAL BODY              INTR058
NEU      CLA     TARG                                         INTR059
         FSB     TABLE                                        INTR060
         TMI     LOOKUP             POSITION EPHEMERIS TAPE   INTR061
         STO     COM+19             T-TO                      INTR062
         SUB     20.                                          INTR063
         TPL     LOOKUP                                       INTR064
         CLA     COM+19                                       INTR065
         FDH     4.                                           INTR066
         STQ     COM+18                                       INTR067
         CLA     COM+18             (T-TO)/4.                 INTR068
         CALL    FIX                                          INTR069
         CALL    FLOAT                                        INTR070
         STO     COM+17                                       INTR071
         CHS                                                  INTR072
         FAD     COM+18                                       INTR073
         STO     COM+1                                        INTR074
         CLA     TARG+1                                       INTR075
         FDH     4.                                           INTR076
         STQ     COM                                          INTR077
         CLA     COM                                          INTR078
         FAD     COM+1                                        INTR079
         STO     TARG+2                                       INTR080
         LDQ     COM+19                                       INTR081
         FMP     =9.                                          INTR082
         CALL    FIX                                          INTR083
         ADD     KERNO+1                                      INTR084
         STA     GG                                           INTR085
         LDQ     COM+17                                       INTR086
         FMP     =54.                                         INTR087
         CALL    FIX                                          INTR088
         ADD     KERNO+2                                      INTR089
         STA     HH                                           INTR090
         AXT     2*SEP,4            MUST INTERPOLATE          INTR091
         STZ     XN+2*SEP,4         SO CLEAR STORAGE          INTR092
         TIX     *-1,4,1                                      INTR093
         AXT     BSEP,4             SET GRAVITATIONAL         INTR094
         STZ     KBO+BSEP,4         COEFFICIENTS              INTR095
         TIX     *-1,4,1            TO ZERO                   INTR096
         LXA     CENTER,4                                     INTR097
         PXD     ,4                                           INTR098
         COM                                                  INTR099
         PDX     ,2                                           INTR100
```


PHILCO
A SUBSIDIARY OF Ford Motor Company
WDL DIVISION

NTR, INTRI (Cont'd)

```
        TXH     HELIO,4,1                                       INTR1010
GEO     AXT     3,1                                             INTR1020
        CLA     GRAV+3,1                                        INTR1030
        STO     KBO+3,1                                         INTR1040
        TIX     *-2,1,1                                         INTR1050
        STZ     KBO-1,2                                         INTR1060
        CLA     EWORTO                                          INTR1070
        STO     EWORT                                           INTR1080
        CLA     HWORTE                                          INTR1090
        STO     HWORT                                           INTR1100
        TXH     GG12,4,0                                        INTR1110
        CLA     RJ                      TEST FOR                INTR1120
        SUB     R                       INCLUSION OF            INTR1130
        TPL     GG12                    JUPITER IF              INTR1140
        CLA     HWORTJ                  EARTH IS                INTR1150
        STO     HWORT                   CENTRAL BODY            INTR1160
        CLA     GRAV+6                                          INTR1170
        STO     KB6                                             INTR1180
        TXL     GG12,,**                                        INTR1190
HELIO   AXT     BSEP,1                                          INTR1200
        CLA     GRAV+BSEP,1                                     INTR1210
        STO     KBO+BSEP,1                                      INTR1220
        TIX     *-2,1,1                                         INTR1230
        STZ     KB2-3,2                                         INTR1240
ORTHO   CLA     EWORTO                                          INTR1250
        STO     EWORT                                           INTR1260
        CLA     HWORTO                                          INTR1270
        STO     HWORT                                           INTR1280
GG12    CLA     VEL                     CHECK                   INTR1290
        TZE     GG6                     FOR VELOCITY            INTR1300
        CLA     EWORTO                  OPTIONS                 INTR1310
        STO     EWORT                                           INTR1320
        CLA     HWORTV                                          INTR1330
        STO     HWORT                                           INTR1340
GG6     CLA     EWORT                                           INTR1350
        TZE     GG+2                                            INTR1360
        CLA     EVEL                                            INTR1370
        STO     VEL+1                                           INTR1380
        CLA     TARG+1                  GEOCENTRIC              INTR1390
        TSX     TAB,4                   INTERPOLATION           INTR1400
GG      PZE     **,,9                                           INTR1410
        PZE     XN,,EWORT+1                                     INTR1420
        CLA     HWORT                                           INTR1430
        TZE     HH+2                                            INTR1440
        CLA     HVEL                                            INTR1450
        STO     VEL+1                                           INTR1460
        CLA     TARG+2                  HELIOCENTRIC            INTR1470
        TSX     TAB,4                   INTERPOLATION           INTR1480
HH      PZE     **,,54                                          INTR1490
        PZE     XN+3,,HWORT+1                                   INTR1500
RESET   AXT     3,4                     PLACE                   INTR1510
```

INTR, INTRI (Cont'd)

```
         CLA    XN+6,4                  COORDINATES                          INTR1520
         FDH    EVEL                    IN OLD                               INTR1530
         STQ    XN+18,4                 FORMAT                               INTR1540
         CLA    XN+9,4                                                       INTR1550
         FDH    EVEL                                                         INTR1560
         STQ    XN+12,4                                                      INTR1570
         LDQ    XN+3,4                                                       INTR1580
         STQ    XN+6,4                                                       INTR1590
         FMP    MU                                                           INTR1600
         FSB    XN+18,4                                                      INTR1610
         STO    XN+9,4                  POSITION OF SUN                      INTR1620
         STZ    XN+3,4                                                       INTR1630
         LDQ    XN+3,4                                                       INTR1640
         STQ    XN+6,4                                                       INTR1650
         FMP    MU                                                           INTR1660
         FSB    XN+18,4                                                      INTR1670
         STO    XN+9,4                  VELOCITY OF SUN                      INTR1680
         STZ    XN+3,4                                                       INTR1690
         TIX    RESET+1,4,1                                                  INTR1700
         LXA    CENTER,4                                                     INTR1710
         LXD    VEL,2                   IX2 NOT ZERO IF VELOCITY OPTION      INTR1720
         TXH    RVPRT,2,0               GET COORDINATES FOR PRINTING         INTR1730
         TXH    TESTM,4,0                                                    INTR1740
         NZT    KB6                     EARTH-CENTERED                       INTR1750
         TRA    FLEE                                                         INTR176
         AXT    3,1                                                          INTR177
         CLA    XN+21,1                                                      INTR178
         FAD    XN+9,1                                                       INTR179
         STO    XN+21,1                                                      INTR180
         TIX    *-3,1,1                                                      INTR181
         TXL    FLEE                                                         INTR182
TESTM    TXH    RVPRT,4,1                                                    INTR183
         AXT    3,1                     MOON-CENTERED                        INTR184
         CLS    XN+6,1                                                       INTR185
         STO    XN+3,1                                                       INTR186
         CLA    XN+9,1                                                       INTR187
         FSB    XN+6,1                                                       INTR188
         STO    XN+9,1                                                       INTR189
         STZ    XN+6,1                                                       INTR190
         TIX    *-6,1,1                                                      INTR191
FLEE     CLA    CENTER                                                       INTR192
         STO    KERNO                                                        INTR193
         CLA    TARG                                                         INTR194
         STO    TARGO                                                        INTR195
         CLA    TARG+1                                                       INTR196
         STO    TARGO+1                                                      INTR197
         AXT    21,1                    TRANSFER POSITIONS                   INTR198
         CLA    XN+21,1                                                      INTR199
OUP      STO    **,1                                                         INTR200
         TIX    OUP-1,1,1                                                    INTR201
         CLA    VEL                                                          INTR202
```

INTR, INTRI (Cont'd)

```
        TZE     OUT                DO NOT TRANSFER VELOCITY       INTR2030
        AXT     21,1                                              INTR2040
        CLA     XN,+21,1           TRANSFER VELOCITIES            INTR2050
(N,V    STO     **,1                                              INTR2060
        TIX     XN,V-1,1,1                                        INTR2070
)UT     LXD     HELIO-1,1                                         INTR2080
        LXD     TRAP+1,2                                          INTR2090
        LXD     TRAP,4                                            INTR2100
        TRA     8,4                                               INTR2110
        REM     TSX TAB,4          (AC)=INTERPOLATIVE ARGUMENT    INTR2120
        REM     PZE B,,K                                          INTR2130
        REM     PZE A,,C                                          INTR2140
        REM                        B=START OF DATA BLOCK          INTR2150
        REM                        K=WORDS PER SUB BLOCK          INTR2160
        REM                        A=START OF RESULT BLOCK        INTR2170
        REM                        C=SKIP CODE WORD LOCATION      INTR2180
'AB     SXD     COM+9,4                                           INTR2190
        SXD     COM+8,2                                           INTR2200
        SXD     COM+7,1                                           INTR2210
        STO     ARG                                               INTR2220
        CLA     1,4                                               INTR2230
        STA     TAB18                                             INTR2240
        LRS     18                                                INTR2250
        ADD     1,4                                               INTR2260
        STA     TAB21                                             INTR2270
        CLA     2,4                                               INTR2280
        PAX     ,1                                                INTR2290
        TXI     *+1,1,SEP                                         INTR2300
        SXA     TAB29,1                                           INTR2310
        ARS     18                                                INTR2320
        STA     VELOP                                             INTR2330
        AXT     2,1                                               INTR2340
        CLA     VEL                                               INTR2350
        TNZ     VELOP                                             INTR2360
        TXI     VELOP,1,-1                                        INTR2370
'ELOP   CLA     **,1               PICK UP SKIP                   INTR2380
        STO     VEL+2              CODE WORD                      INTR2390
        TXL     POSOP,1,1                                         INTR2400
        REM                        VELOCITY OPTION                INTR2410
        CLA     ARG                                               INTR2420
        TSX     COEFF.,4           FORM THE                       INTR2430
        AXT     3,4                E1(2J).                        INTR2440
        CLS     COM+13,4                                          INTR2450
        STO     COM+16,4                                          INTR2460
        TIX     *-2,4,1                                           INTR2470
        CLA     1.                 FORM THE                       INTR2480
        FSB     ARG                E0(2J).                        INTR2490
        TSX     COEFF.,4                                          INTR2500
        TRA     TAB1-1                                            INTR2510
        REM                        POSITION OPTION                INTR2520
OSOP    CLA     ARG                                               INTR2530
```

INTR, INTRI (Cont'd)

```
         TSX    COEFF,4             FORM THE              INTR254
         AXT    3,4                 E1(2J)                INTR255
         CLA    COM+13,4                                  INTR256
         STO    COM+16,4                                  INTR257
         TIX    *-2,4,1                                   INTR258
         CLA    1.                  FORM THE              INTR259
         FSB    ARG                 E0(2J)                INTR260
         TSX    COEFF,4                                   INTR261
         LXA    TAB29,4                                   INTR262
         TXI    *+1,4,-SEP                                INTR263
         SXA    TAB29,4                                   INTR264
         SXD    COM+4,1                                   INTR265
TAB1     AXT    0,3                                       INTR266
         LDQ    VEL+2                                     INTR267
         PXD                                              INTR268
         LGL    2                                         INTR269
         PAX    ,4                                        INTR270
         STQ    VEL+2                                     INTR271
         CAL    =03                                       INTR272
         ORS    VEL+2                                     INTR273
         TXH    END,4,2                                   INTR274
         TXH    FORT,4,0                                  INTR275
         TXI    *+1,1,-3                                  INTR276
         TXI    TAB1+1,2,-1                               INTR277
FORT     AXT    3,4                                       INTR278
TAB18    LDQ    **,1                X(0)                  INTR279
         FMP    COM+13,4                                  INTR280
         STO    COM                                       INTR281
TAB21    LDQ    **,1                X(1)                  INTR282
         FMP    COM+16,4                                  INTR283
         FAD    COM                                       INTR284
         STO    COM+4,4                                   INTR285
         TXI    *+1,1,-1                                  INTR286
         TIX    *-8,4,1                                   INTR287
         FAD    COM+2                                     INTR288
         FAD    COM+1                                     INTR289
TAB29    STO    **,2                X(T)                  INTR290
         TXI    TAB1+1,2,-1                               INTR291
END      LXD    COM+4,1                                   INTR292
         TIX    VELOP,1,1                                 INTR293
         LXD    COM+9,4                                   INTR294
         LXD    COM+8,2                                   INTR295
         LXD    COM+7,1                                   INTR296
         TRA    3,4                                       INTR297
COEFF    STO    COM+10              CALCULATE             INTR298
         LDQ    COM+10              POSITION              INTR299
         FMP    COM+10              COEFFICIENTS          INTR300
         STO    COM+12              FOR EVERETT,S         INTR301
         FSB    1.                  INTERPOLATION         INTR302
         FDH    6.                  FORMULA               INTR303
         FMP    COM+10                                    INTR304
```

INTR, INTRI (Cont'd)

```
        STO     COM+11                                          INTR3050
        CLA     COM+12                                          INTR3060
        FSB     4.                                              INTR3070
        FDH     20.                                             INTR3080
        FMP     COM+11                                          INTR3090
        STO     COM+12                                          INTR3100
        TRA     1,4                                             INTR3110
COEFF.  STO     COM              CALCULATE                      INTR3120
        CLS     1.               VELOCITY                       INTR3130
        FDH     VEL+1            COEFFICIENTS                   INTR3140
        STQ     COM+10           FOR EVERETT.S                  INTR3150
        LDQ     COM              INTERPOLATION                  INTR3160
        FMP     COM              FORMULA                        INTR3170
        STO     COM+12                                          INTR3180
        XCA                                                     INTR3190
        FMP     3.                                              INTR3200
        FSB     1.                                              INTR3210
        FDH     6.                                              INTR3220
        FMP     COM+10                                          INTR3230
        STO     COM+11                                          INTR3240
        CLA     COM+12                                          INTR3250
        FSB     3.                                              INTR3260
        XCA                                                     INTR3270
        FMP     5.                                              INTR3280
        XCA                                                     INTR3290
        FMP     COM+12                                          INTR3300
        FAD     4.                                              INTR3310
        FDH     120.                                            INTR3320
        FMP     COM+10                                          INTR3330
        STO     COM+12                                          INTR3340
        TRA     1,4                                             INTR3350
RVPRT   AXT     0,2              EXPRESS ALL                    INTR3360
        AXT     3,5              BODIES GEOCENTRICALLY          INTR3370
        CLA     XN+9,2                                          INTR3380
        FAD     XN+9,4                                          INTR3390
        STO     XN+9,2                                          INTR3400
        CLA     VEL                                             INTR3410
        TZE     *+4                                             INTR3420
        CLA     XN,+9,2                                         INTR3430
        FAD     XN,+9,4                                         INTR3440
        STO     XN,+9,2                                         INTR3450
        TXI     *+1,2,-1                                        INTR3460
        TIX     RVPRT+2,4,1                                     INTR3470
        TXH     RVPRT+1,2,9-SEP                                 INTR3480
        CLA     CENTER           BUFFER                         INTR3490
        ALS     1                CENTRAL                        INTR3500
        ADD     CENTER           BODY                           INTR3510
        PAC     ,4                                              INTR3520
        CLA     XN,4                                            INTR3530
        STO     COM+3,1                                         INTR3540
        CLA     XN,,4                                           INTR3550
```

INTR, INTRI (Cont'd)

```
        STO     COM+6,1                                         INTR356
        TXI     *+1,4,-1                                        INTR357
        TIX     *-5,1,1                                         INTR358
RVPRT1  AXT     0,2                 EXPRESS ALL                 INTR359
        AXT     3,4                 BODIES IN TERMS             INTR360
        CLA     XN,2                OF THE CENTRAL              INTR361
        FSB     COM+3,4             BODY                        INTR362
        STO     XN,2                                            INTR363
        CLA     VEL                                             INTR364
        TZE     *+4                                             INTR365
        CLA     XN,,2                                           INTR366
        FSB     COM+6,4                                         INTR367
        STO     XN,,2                                           INTR368
        TXI     *+1,2,-1                                        INTR369
        TIX     RVPRT1+2,4,1                                    INTR370
        TXH     RVPRT1+1,2,-SEP                                 INTR371
        TXL     FLEE                                            INTR372
LOOKUP  CLA     TLAST                                           INTR373
        FSB     TARG                                            INTR374
        TNZ     *+2                                             INTR375
        TRA     ERR1                                            INTR376
        TMI     *-1                                             INTR377
        CLA     TARG                                            INTR378
        FSB     TFIRST                                          INTR379
        TPL     *+2                                             INTR380
        TRA     ERR1                                            INTR381
        FDH     20.                                             INTR382
        XCA                                                     INTR383
        CALL    FIX                                             INTR384
        CALL    FLOAT                                           INTR385
        XCA                                                     INTR386
        FMP     20.                                             INTR387
        FAD     TFIRST                                          INTR388
        STO     COM+1               TIME ON RECORD              INTR389
        CLA     TABLE                                           INTR390
        FSB     COM+1                                           INTR391
        TMI     FINDIT                                          INTR392
        FDH     20.                                             INTR393
        XCA                                                     INTR394
        CALL    FIX                                             INTR395
        ADD     1F                  RECORDS TO                  INTR396
        PAX     ,4                  BE BACKED                   INTR397
        TEFA    *+1                                             INTR398
        TXL     *+3,4,20            OVER                        INTR399
        REWA    6                                               INTR400
        TRA     FINDIT                                          INTR401
        BSRA    6                                               INTR402
        TIX     *-1,4,1                                         INTR403
FINDIT  AXT     10,2                                            INTR404
        SDHA    6                                               INTR405
        RTBA    6                                               INTR406
```

INTR, INTRI (Cont'd)

```
        RCHA    IO                                                  INTR4070
        TCOA    *                                                   INTR4080
        TEFA    ERR1                                                INTR4090
        CLA     TABLE                                               INTR4100
        SUB     COM+1                                               INTR4110
        TMI     *+2                                                 INTR4120
        TNZ     LOOKUP              TAPE NOT POSITIONED PROPERLY    INTR4130
        TNZ     FINDIT                                              INTR4140
        AXT     513,4                                               INTR4150
        CAL     TABLE               CHECK                           INTR4160
        ACL     A+513,4             SUM                             INTR4170
        TIX     *-1,4,1                                             INTR4180
        TRCA    BSRA                                                INTR4190
        LAS     C                                                   INTR4200
        TRA     *+2                                                 INTR4210
        TRA     SCALE                                               INTR4220
BSRA    BSRA    6                                                   INTR4230
        TIX     FINDIT+1,2,1                                        INTR4240
        CALL    ERP                                                 INTR4250
IO      IOCD    TABLE,,515                                          INTR4260
ERR1    CALL    ERPT                                                INTR4270
        PZE     TARG                                                INTR4280
        REM                         SCALING                         INTR4290
        REM                         LOOP                            INTR4300
SCALE   AXT     189,4                                               INTR4310
        LDQ     A+189,4             SCALE                           INTR4320
        FMP     SCALE1              GEOCENTRIC                      INTR4330
        STO     A+189,4             EPHEMERIS                       INTR4340
        TIX     *-3,4,1                                             INTR4350
        AXT     324,4                                               INTR4360
        LDQ     B+324,4             SCALE                           INTR4370
        FMP     SCALE2              HELIOCENTRIC                    INTR4380
        STO     B+324,4             EPHEMERIS                       INTR4390
        TIX     *-3,4,1                                             INTR4400
        CLA     GRAV                COMPUTE                         INTR4410
        FAD     GRAV+1              MASS RATIO                      INTR4420
        STO     COM                 OF MOON                         INTR4430
        CLA     GRAV+1              TO EARTH-MOON                   INTR4440
        FDH     COM                 BARYCENTER                      INTR4450
        STQ     MU                  FOR POSITION OF EARTH           INTR4460
        TRA     NEU                                                 INTR4470
SCALE1  DEC     6378.150            EARTH RADIUS                    INTR4480
SCALE2  DEC     149598500.          ASTRONOMICAL UNIT (JPL, JULY 1961INTR4490
TFIRST  DEC     3892.0              0 HR AUG 28, 1960 JD = 2437174.5 INTR4500
TLAST   DEC     7292.0              0 HR DEC 19, 1969 JD = 2440554.5 INTR4510
TRAP    PZE                                                         INTR4520
        PZE                                                         INTR4530
RJ      DEC     1E6                 JUPITER TEST DISTANCE           INTR4540
TEMPDT  DEC     34,                 E.T.-U.T.                       INTR4550
        PZE                                                         INTR4560
GRAV    DEC     3.98602E5           EARTH                           INTR4570
```

INTR, INTRI (Cont'd)

```
        DEC     4.8983349E3          MOON                              INTR458
        DEC     1.3252312E11         SUN                               INTR459
        DEC     3.2429889E5          VENUS                             INTR460
        DEC     4.2915518E4          MARS                              INTR461
        DEC     0                    REMOVE BARYCENTER                 INTR462
        DEC     1.2652701E8          JUPITER                           INTR463
EVEL    DEC     86400.                                                 INTR464
HVEL    DEC     345600.                                                INTR465
MU      PZE                          MASS RATIO OF MOON TO BARYCENTER  INTR466
        OCT     000000520052         MARS, JUPITER VELOCITY            INTR467
HWORT   OCT     0                                                      INTR468
        OCT     527777777777         MOON VELOCITY                     INTR469
EWORT   OCT     0                                                      INTR470
HWORTJ  OCT     000000005252         BARYCENTER, JUPITER               INTR471
HWORTE  OCT     000000005200         BARYCENTER                        INTR472
EWORTO  OCT     527777777777                                           INTR473
HWORTO  OCT     000052525252                                           INTR474
HWORTV  OCT     525252525252                                           INTR475
TARGO   DEC     20,,0                FORMER TIME                       INTR476
KERNO   PZE                          FORMER CENTER                     INTR477
        PZE     A                    GEOCENTRIC REFERENCE              INTR478
        PZE     B                    HELIOCENTRIC REFERENCE            INTR479
ARG     PZE     0                                                      INTR480
1F      DEC     1                                                      INTR481
1.      DEC     1.                                                     INTR482
3.      DEC     3.                                                     INTR483
4.      DEC     4.                                                     INTR484
5.      DEC     5.                                                     INTR485
6.      DEC     6.                                                     INTR486
20.     DEC     20.                                                    INTR487
120.    DEC     120.                                                   INTR488
86400.  DEC     86400.                                                 INTR489
VEL     BSS     3                    VELOCITY OPTION,H,SKIP CODE       INTR490
*                                                                      INTR491
TABLE   DEC     0                    RESERVE                           INTR492
A       BSS     189                  FOR                               INTR493
B       BSS     324                  WORKING                           INTR494
C       BSS     1                    EPHEMERIS                         INTR495
*                                                                      INTR496
TARG    PZE                                                            INTR497
        PZE                                                            INTR498
        PZE                                                            INTR499
CENTER  PZE                                                            INTR500
COM     BSS     21                                                     INTR501
VAFLG   PZE                                                            INTR502
FQFLG   BSS     3                                                      INTR503
KB0     PZE                                                            INTR504
KB1     PZE                                                            INTR505
KB2     PZE                                                            INTR506
KB3     PZE                                                            INTR507
KB4     PZE                                                            INTR508
```

INTR, INTRI (Cont'd)

```
KB5     PZE                                    INTR5090
KB6     PZE                                    INTR5100
XN      BSS     21                             INTR5110
XN.     BSS     21                             INTR5120
T       PZE                                    INTR5130
        PZE                                    INTR5140
R       PZE                                    INTR5150
BSEP    SYN     7                              INTR5160
SEP     SYN     XN.-XN                         INTR5170
        END                                    INTR
```

Subroutine: INV3

Purpose: To invert a matrix of any dimension up to a 6 by 6.

Calling Sequence:

CALL INV3 (A, N, DETERM)

Input and **Output**

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
IO	A	6,6			A(I) Matrix to be inverted
					A(O) Inverse of input matrix
I	N	1			Dimension of input matrix
O	DETERM	1			Matrix determinant

Common storages used or required: None

Subroutines required: None

Functions required: ABS

Approximate number of storages required:

INV3

```
*       LABEL                                                           INV3
C       SUBROUTINE FOR INVERTING SQUARE MATRICES WHICH ARE 6 BY 6 OR LESS INV30000
C       (TO INVERT LARGER MATRICES,SAY M BY M, DIMENSION IPIVOT(M),A(M*M),INV30010
C       INDEX(M,2),PIVOT(M) AND RECOMPILE)                              INV30020
C       A IS THE SQUARE MATRIX TO BE INVERTED                           INV30030
C       N IS THE SIZE OF A (A IS AN N BY N MATRIX)                      INV30040
C       THE SUBROUTINE RETURNS A INVERSE IN PLACE OF A AND THE DETERMINANTINV30050
C       IN DETERM.                                                      INV30060
        SUBROUTINE INV3(A,N,DETERM)                                     INV30070
        DIMENSION IPIVOT(6),A(36) ,INDEX(6,2),PIVOT(6)                  INV30080
        EQUIVALENCE (IROW,JROW), (ICOLUM,JCOLUM),(AMAX,T,SWAP)          INV30090
        DETERM=1.0                                                      INV30100
        DO 20  J=1,N                                                    INV30110
 20     IPIVOT(J)=0                                                     INV30120
        DO 550  I=1,N                                                   INV30130
        AMAX=0.0                                                        INV30140
        DO 105 J=1,N                                                    INV30150
        IF(IPIVOT(J)-1) 60,105,60                                       INV30160
 60     DO 100 K=1,N                                                    INV30170
        IF(IPIVOT(K)-1) 80,100,740                                      INV30180
   80   M = N*(K-1)+J                                                   INV30190
        IF (ABSF(AMAX)-ABSF(A(M))) 85,100,100                           INV30200
 85     IROW=J                                                          INV30210
        ICOLUM=K                                                        INV30220
        AMAX = A(M)                                                     INV30230
 100    CONTINUE                                                        INV30240
 105    CONTINUE                                                        INV30250
        IPIVOT(ICOLUM)=IPIVOT(ICOLUM)+1                                 INV30260
        IF(IROW-ICOLUM) 140,260,140                                     INV30270
 140    DETERM=-DETERM                                                  INV30280
        DO 200 L=1,N                                                    INV30290
        M = N*(L-1)                                                     INV30300
        M1 = M+ICOLUM                                                   INV30310
        M = M+IROW                                                      INV30320
        SWAP = A(M)                                                     INV30330
        A(M) = A(M1)                                                    INV30340
  200   A(M1) = SWAP                                                    INV30350
 260    INDEX(I,1)=IROW                                                 INV30360
        INDEX(I,2)=ICOLUM                                               INV30370
        M = N*(ICOLUM-1)+ICOLUM                                         INV30380
        PIVOT(I) = A(M)                                                 INV30390
        DETERM=DETERM*PIVOT(I)                                          INV30400
        A(M) = 1.0                                                      INV30410
        DO 350 L=1,N                                                    INV30420
        M = N*(L-1)+ICOLUM                                              INV30430
  350   A(M) = A(M)/PIVOT(I)                                            INV30440
        DO 550 L1=1,N                                                   INV30450
        IF(L1-ICOLUM) 400,550,400                                       INV30460
  400   M = N*(ICOLUM-1)+L1                                             INV30470
        T = A(M)                                                        INV30480
        A(M) = 0.                                                       INV30490
```

INV3 (Cont'd)

```
      DO 450  L=1,N                                          INV3050
      M = N*(L-1)                                            INV3051
      M1 = M+ICOLUM                                          INV3052
      M = M+L1                                               INV3053
  450 A(M) = A(M)-A(M1)*T                                    INV3054
  550 CONTINUE                                               INV3055
      DO 710 I=1,N                                           INV3056
      L=N+1-I                                                INV3057
      IF(INDEX(L,1)-INDEX(L,2)) 630,710,630                  INV3058
  630 JROW=INDEX(L,1)                                        INV3059
      JCOLUM=INDEX(L,2)                                      INV3060
      M = N*(JROW-1)                                         INV3061
      M1 = N*(JCOLUM-1)                                      INV3062
      DO 705 K=1,N                                           INV3063
      M = M+1                                                INV3064
      M1 = M1+1                                              INV3065
      SWAP = A(M)                                            INV3066
      A(M) = A(M1)                                           INV3067
      A(M1) = SWAP                                           INV3068
  705 CONTINUE                                               INV3069
  710 CONTINUE                                               INV3070
  740 RETURN                                                 INV3071
      END                                                    INV3
```

Page Intentionally Left Blank

Subroutine: INVAO

Purpose: To form the inverse of a transition matrix for a linear, conservative dynamic system. If Φ is a partioned matrix such as

$$\Phi = \begin{pmatrix} \Phi_1 & \Phi_2 \\ \Phi_3 & \Phi_4 \end{pmatrix} \quad \text{then INVAO computes } \Phi^{-1} = \begin{pmatrix} \Phi_4^T & -\Phi_2^T \\ -\Phi_3^T & \Phi_1^T \end{pmatrix}$$

Calling Sequence:

CALL INVAO (AO, AOI)

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	AO	6,6	$\Phi(t_2,t_1)$		Transition Matrix
O	AO1	6,6	$\Phi^{-1}(t_2,t_1)$		Inverse Transition Matrix

Common storages used or required: None

Subroutines required: None

Functions required: None

Approximate number of storages required:

INVAO

```
*       LABEL                                                         INVA
*       SYMBOL TABLE                                                  INVA
        SUBROUTINE INVAO(AO,AOI)                                      INVA0000
        DIMENSION AO(6,6),AOI(6,6)                                    INVA0010
        DO 1 I=1,3                                                    INVA0020
        II=I+3                                                        INVA0030
        DO 1 J=1,3                                                    INVA0040
        JJ=J+3                                                        INVA0050
        AOI(I,J)=AO(JJ,II)                                            INVA0060
        AOI(II,J)=-AO(JJ,I)                                           INVA0070
        AOI(II,JJ)=AO(J,I)                                            INVA0080
    1   AOI(I,JJ)=-AO(J,II)                                           INVA0090
        RETURN                                                        INVA0100
        END                                                           INVA
```

Subroutine: LOADO

Purpose: To transfer the transition matrix from its storage in the integration package's T block to the matrix in the call list.

Calling Sequence:

CALL LOADO (AO)

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
O	AO	(6,6)	$\Phi(t_2 t_1)$		Transition Matrix

Common storages used or required: T

Subroutines required: None

Functions required: None

Approximate number of storages required:

LOADO

```
*     LABEL                                                        LOAD
*     SYMBOL TABLE                                                 LOAD
      SUBROUTINE LOADO(AC)                                         LOAD0000
      COMMON T                                                     LOAD0010
      DIMENSION T(1360),AO(6,6)                                    LOAD0020
      K=6                                                          LOAD0030
      JJ=27                                                        LOAD0040
      DO1 I=1,6                                                    LOAD0050
      DO 1 J=1,3                                                   LOAD0060
      K=K+1                                                        LOAD0070
      KK=J+3                                                       LOAD0080
      JJ=JJ+1                                                      LOAD0090
      AO(KK,I)=T(JJ)                                               LOAD0100
      AO(J,I)=T(K)                                                 LOAD0110
    1 CONTINUE                                                     LOAD0120
      RETURN                                                       LOAD0130
      END                                                          LOAD
```

Subroutine: LOADT

Purpose: To place unit initial conditions in the T block for the variational equations which are being used to generate the transition matrix.

Calling Sequence:

CALL LOADT

Input and **Output**

Common storages used or required: T

Subroutines required: None

Functions required: None

Approximate number of storages required:

LOADT

```
*       LABEL                                                   LOAT
*       SYMBOL TABLE                                            LOAT
        SUBROUTINE LOADT                                        LOAT0000
C       SUBROUTINE LOADT PUTS UNIT ICS ON PERTURBATION EQUATIONS LOAT0010
        COMMON T                                                LOAT0020
        DIMENSION T(1360),X(3,6),V(3,6)                         LOAT0030
        EQUIVALENCE (T(7), X),(T(28),V)                         LOAT0040
        DO 2 I=1,3                                              LOAT0050
        L=I+3                                                   LOAT0060
        DO 1 J=1,3                                              LOAT0070
        K=J+3                                                   LOAT0080
        X(I,J)=0.                                               LOAT0090
        V(I,K)=0.                                               LOAT0100
        X(I,K)=0.                                               LOAT0110
        V(I,J)=0.                                               LOAT0120
      1 CONTINUE                                                LOAT0130
        X(I,I)=1.                                               LOAT0140
        V(I,L)=1.                                               LOAT0150
      2 CONTINUE                                                LOAT0160
        RETURN                                                  LOAT0170
        END                                                     LOAT
```

Subroutine: MASS

Purpose: To find the relevant gravitational constants to be used in computing planetary perturbations for a given central body. MASS also chooses the initial integration step size as a function of central body.

Calling Sequence:

CALL MASS (NOR, UM, VKB, X, ACCP)

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	NOR	1			Central Body Indicator.
					1=Earth; 2, Moon; 3, Sun; 4,
					Venus; 5, Mars; 6, Jupiter
I	UM	6	μ_i	Km^3/sec^2	Gravitational constants, arranged
					as above
O	VKB	6	μ_i	Km^3/sec^2	μ_i for bodies
I	X	3	X,Y,Z	Km	Position coordinates
O	ACCP	1		seconds	Initial integration step size

Common storages used or required: None

Subroutines required: None

Functions required: FNORM

Approximate number of storages required:

Method of Establishing Integration Step Size and Perturbing Bodies

Central Body	Integration Step Size	Other Bodies to be Considered
Earth	60.0	Moon, Sun. If $\lvert X \rvert \geq 10^6$Km, Jupiter.
Moon	60.0	Earth, Sun.
Sun	43200.0	All
Venus, Mars, Jupiter	60.0	All

The bodies which are not to be used in calculation of pertubation accelerations are eliminated from consideration by placing zeros in array VKB(6). For the bodies which are being considered, the appropriate gravitational constant is placed in VKB(6).

MASS

```
*      LABEL
*      SYMBOL TABLE
       SUBROUTINE MASS(NOR,UM,VKB,X,ACCP)
       DIMENSION UM(6),VKB(6),X(3)
       N = NOR
       DO 1 I=1,6
     1 VKB(I) = 0.
       GO TO (2,3,6,4,4,4),N
     2 VKB(2) = UM(2)
       ACCP = 60.
       DIS = FNORM(X)
       VKB(3) = UM(3)
       IF (DIS-1000000.) 11,10,10
    10 VKB(6) = UM(6)
       GO TO 12
    11 VKB(6) = 0.
    12 RETURN
     3 VKB(1) = UM(1)
       VKB(3) = UM(3)
       ACCP = 60.
       RETURN
     6 ACCP = 43200.
       GO TO 7
     4 ACCP = 60.
     7 DO 5 I=1,6
     5 VKB(I) = UM(I)
       VKB(N) = 0.
       RETURN
       END
```

Page Intentionally Left Blank

Subroutine: MATRX

Purpose: To perform matrix multiplications of matricies with any dimensions up to maximum dimensions of 10 by 10. A zero in the call sequence (J) yields an output matrix $C = A \cdot B$. A one in the call sequence (J) yields an output matrix $C = A \cdot B \cdot A^T$. The matrix products are obtained in double precision.

Calling Sequence:

CALL MATRX (A, B, C, NRA, NCA, NCB, J)

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	A	(10,10)			Input Matrix
I	B	(10,10)			Input Matrix
O	C	(10,10)			Output Matrix
I	NRA	1			Number of Rows of A
I	NCA	1			Number of Columns of A
I	NCB	1			Number of Columns of B
I	J	1			Type of multiplication desired

Common storages used or required: None

Subroutines required: None

Functions required: None

Approximate number of storages required:

```
*       LABEL                                                           MATX
*       SYMBOL TABLE                                                    MATX
CEC2000  SUBROUTINE MATRX                                               MATX
        SUBROUTINE MATRX(A,B,C,NRA,NCA,NCB,J)                           MATX0000
C       J=0 A*B=C                                                       MATX0010
C       J=1 A*B*AT=C  T MEANS TRANSPOSE                                 MATX0020
C       NRA=NUMBER OF ROWS OF A                                         MATX0030
C       NCA=COLUMNS OF A                                                MATX0040
C       NCB=COLUMNS OF B                                                MATX0050
C       ACCUMULATION OF PRODUCTS IN DOUBLE PRECISION                    MATX0060
        DIMENSION A(100),B(100),C(100),D(100)                           MATX0080
D       DUD=0.                                                          MATX0090
D       DUM=0.                                                          MATX0100
        DO 1 I=1,NRA                                                    MATX0110
        DO 1 K=1,NCB                                                    MATX0120
        NC=I+(K-1)*NRA                                                  MATX0130
D       CRUD=0.                                                         MATX0140
        DO 2 L=1,NCA                                                    MATX0150
        NA=I+(L-1)*NRA                                                  MATX0160
        NB=L+(K-1)*NCA                                                  MATX0170
        DUD=A(NA)                                                       MATX0180
        DUM=B(NB)                                                       MATX0190
D       CRUD=CRUD+DUD*DUM                                               MATX0200
      2 CONTINUE                                                        MATX0210
        D(NC)=CRUD                                                      MATX0220
      1 CONTINUE                                                        MATX0230
        IF(J)3,3,4                                                      MATX0240
      3 CONTINUE                                                        MATX0250
        KK=NCB*NRA                                                      MATX0260
        DO 5 I=1,KK                                                     MATX0270
      5 C(I)=D(I)                                                       MATX0280
        GO TO 10                                                        MATX0290
      4 CONTINUE                                                        MATX0300
        DO 6 I=1,NRA                                                    MATX0310
        DO 6 K=1,NRA                                                    MATX0320
        NC=I+(K-1)*NRA                                                  MATX0340
D       CRUD=0.                                                         MATX0350
        DO 7 L=1,NCA                                                    MATX0360
        NA=I+(L-1)*NRA                                                  MATX0370
        NB=K+(L-1)*NRA                                                  MATX0380
        DUM=D(NA)                                                       MATX0390
        DUD=A(NB)                                                       MATX0400
D       CRUD=CRUD+DUM*DUD                                               MATX0410
      7 CONTINUE                                                        MATX0420
        C(NC)=CRUD                                                      MATX0430
      6 CONTINUE                                                        MATX0440
     10 CONTINUE                                                        MATX0510
        RETURN                                                          MATX0520
        END                                                             MATX
```

Subroutine: MATSUB

Purpose: The subroutine is primarily logic which controls (1) trajectory and data printout, (2) updating of the state covariance matrix for observations by calling subroutines EARTR, ONBTR, and MONBTR, and (3) updating of the guidance covariance matrix by calling subroutine GUID.

Calling Sequence:

CALL MATSUB

Input and Output

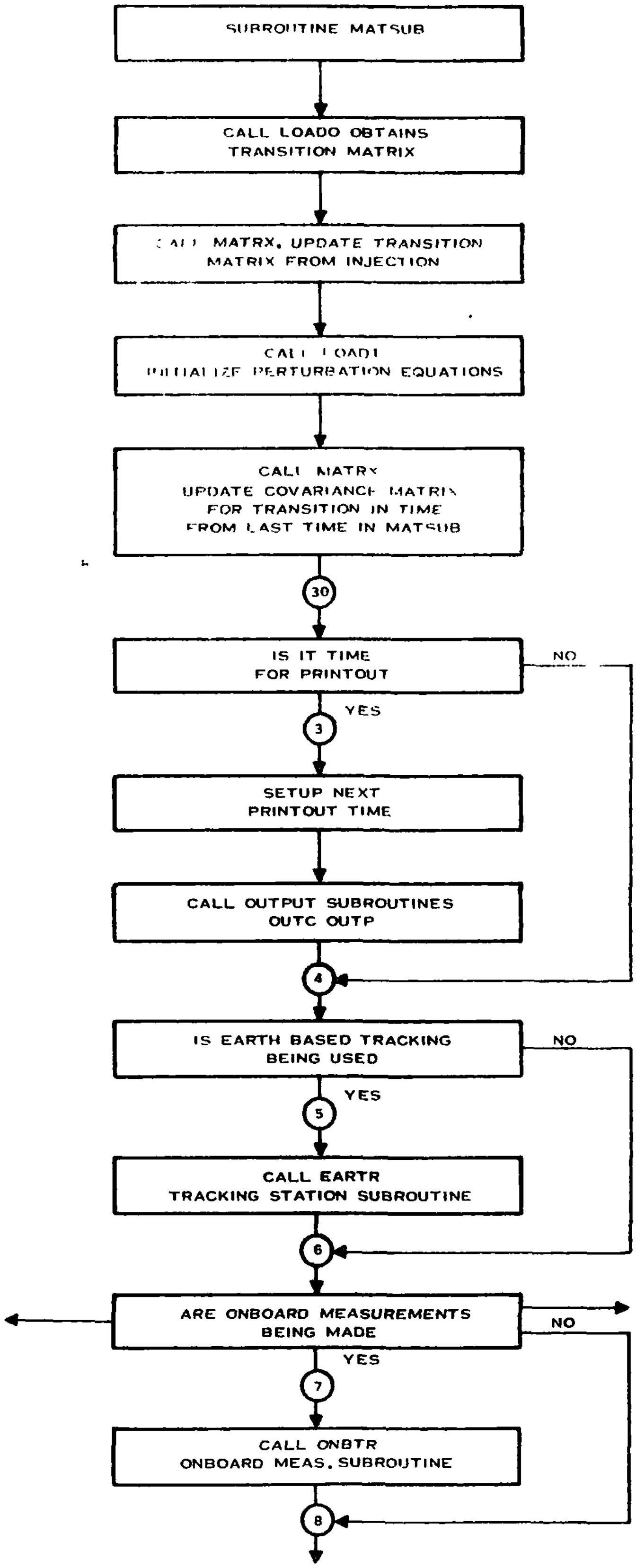
I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I					Quantities obtained from common
O					Quantities returned to common

Common storages used or required: T, S, C, IC

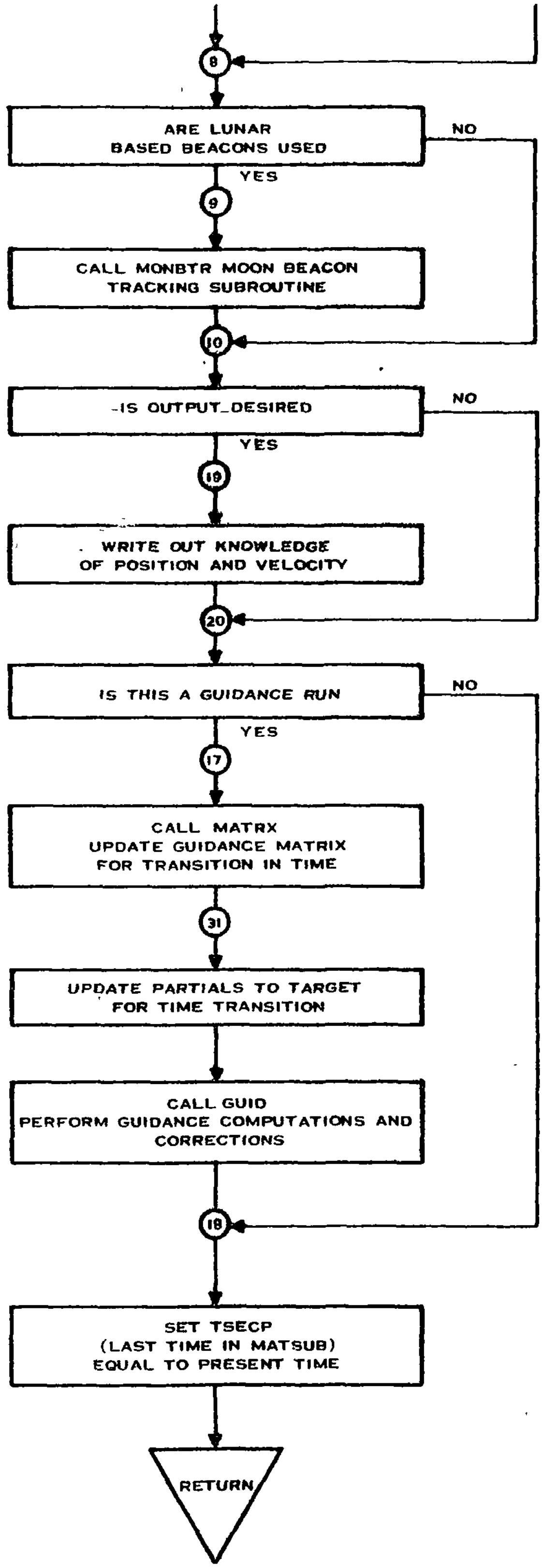
Subroutines required: EARTR, GUID, INVAO, LOADO, LOADT, MATRX, MONBTR, ONBTR, OUTC, OUTP

Functions required: SQRT (FIL)(STH)

Approximate number of storages required: 346 DEC


SUBROUTINE MATSUB
CALL LOADO OBTAINS TRANSITION MATRIX
CALL MATRX, UPDATE TRANSITION MATRIX FROM INJECTION
CALL LOAD1 INITIALIZE PERTURBATION EQUATIONS
CALL MATRX UPDATE COVARIANCE MATRIX FOR TRANSITION IN TIME FROM LAST TIME IN MATSUB
30
IS IT TIME FOR PRINTOUT
NO
YES
3
SETUP NEXT PRINTOUT TIME
CALL OUTPUT SUBROUTINES OUTC OUTP
4
IS EARTH BASED TRACKING BEING USED
NO
YES
5
CALL EARTR TRACKING STATION SUBROUTINE
6
ARE ONBOARD MEASUREMENTS BEING MADE
NO
YES
7
CALL ONBTR ONBOARD MEAS. SUBROUTINE
8

MATSUB -2


8
ARE LUNAR BASED BEACONS USED
NO
YES
9
CALL MONBTR MOON BEACON TRACKING SUBROUTINE
10
IS OUTPUT DESIRED
NO
YES
19
WRITE OUT KNOWLEDGE OF POSITION AND VELOCITY
20
IS THIS A GUIDANCE RUN
NO
YES
17
CALL MATRX UPDATE GUIDANCE MATRIX FOR TRANSITION IN TIME
31
UPDATE PARTIALS TO TARGET FOR TIME TRANSITION
CALL GUID PERFORM GUIDANCE COMPUTATIONS AND CORRECTIONS
18
SET TSECP (LAST TIME IN MATSUB) EQUAL TO PRESENT TIME
RETURN

```
*       LABEL                                                                   MATS
*       SYMBOL TABLE                                                            MATS
CEC2006  SUBROUTINE MATSUB                                                      MATS
      SUBROUTINE MATSUB                                                         MATS0000
      COMMON T,S,C,IC                                                           MATS0010
      DIMENSION T(1360),S(1000),C(1000),IC(1),AN(3,3),PO(22),VE(22),            MATS0020
     1EM(3,3),XP(3),VXP(3),EA(3,3),XED(3),VED(3),AO(6,6),P(6,6),                MATS0030
     2TFTA(3,6),PFTA(6,6),AOS(6,6),AOI(6,6)                                     MATS0040
      EQUIVALENCE (T ,TDUM),(S ,SDUM),(C ,CDUM),                                MATS0050
     1(C(138),AN),(C(13),TP1),(C(14),TP2),(C(62),PO),(C(84),VE),                MATS0060
     2(C(120),EM),(C(650),TPRINT),(C(651),TSTRT),(C(30),TSEC),                  MATS0070
     3(C(15),XP),(C(18),VXP),(C(10),TW),(C(11),TF) ,(C(129),EA),                MATS0080
     4(C(652),P),(C(752),AOS),(C(850),AOI)                                      MATS0090
     5,(C(460),PFTA),(C(568),TFTA),(C(647),TOUT)                                MATS0100
      EQUIVALENCE (IC(2),IOR),(IC(7),KOUT),(IC(6),ITARG),(IC(9),LPRINT),MATS0110
     1(IC(190),LSTAT),(IC(191),LONB),(IC(192),LMB),(IC(193),IGUID),             MATS0120
     2(IC(214),NOUT),(C(649),TSECP)                                             MATS0130
      CALL SETN(NIN,NUTS)                                                       MATS0131
      NOUT=1                                                                    MATS0140
      CALL LOADO(AO)                                                            MATS0145
      CALL MATRX(AO,AOS,AOS,6,6,6,0)                                            MATS0150
      CALL LOADT                                                                MATS0155
      CALL MATRX(AO,P,P,6,6,6,1)                                                MATS0160
      DO 30 I=1,6                                                               MATS0165
      DO 30 J=I,6                                                               MATS0170
      P(I,J)=(P(I,J)+P(J,I))/2.                                                 MATS0175
      P(J,I)=P(I,J)                                                             MATS0180
   30 CONTINUE                                                                  MATS0185
      IF(TOUT-TSEC) 3,3,4                                                       MATS0220
    3 CONTINUE                                                                  MATS0230
      TOUT=TOUT+TPRINT                                                          MATS0240
      NOUT=2                                                                    MATS0250
C     NOUT IS USED TO INDICATE OTHER OUTPUT WANTED WHEN = 2                     MATS0260
      CALL OUTC                                                                 MATS0270
      CALL OUTP                                                                 MATS0280
    4 CONTINUE                                                                  MATS0290
C     LSTAT IS SET 2 BY INPUT CARDS IF ANY EARTH BASED TRACKING                 MATS0300
      LSTAT=LSTAT                                                               MATS0310
      GO TO (6,5),LSTAT                                                         MATS0320
    5 CONTINUE                                                                  MATS0330
      CALL EARTR                                                                MATS0340
C     EARTR UDDATS COV MATRIX P FOR EARTH BASED TRACKING                        MATS0350
    6 CONTINUE                                                                  MATS0360
      LONB=LONB                                                                 MATS0370
      GO TO (8,7),LONB                                                          MATS0380
C     LONB IS SET 2 BY INPUT CARDS IF ANY ONBOARD MEASUREMENTS MADE             MATS0390
    7 CONTINUE                                                                  MATS0400
C     ONBTR  UPDATES COV MATRIX P FOR THE ONBOARD MEASUREMENTS                  MATS0410
      CALL ONBTR                                                                MATS0420
    8 CONTINUE                                                                  MATS0430
      LMB=LMB                                                                   MATS0440
```

MATSUB (Cont'd)

```
      GO TO (10,9),LMB                                                  MATS045
C     LMB IS SET 2 BY INPUT CARDS IF MOON BASED BEACONS USED            MATS046
    9 CONTINUE                                                          MATS047
      CALL MONBTR                                                       MATS048
C    MONBTR UPDATS COV MATRIX P FOR BEACONS ON MOON                     MATS049
   10 CONTINUE                                                          MATS050
      GO TO (20,19), NOUT                                               MATS051
  19  CONTINUE                                                          MATS052
      RMSP=SQRTF (P (1,1)+P (2,2)+P (3,3))                              MATS053
      RMSVP=SQRTF(P (4,4)+P (5,5)+P (6,6))                              MATS054
      WRITE OUTPUT TAPE NUTS,700,RMSP,RMSVP                             MATS055
 700  FORMAT(42H KNOWLEDGE OF STATE AFTER ALL OBSERVATIONS,/            MATS056
     124H              RMS POSITION=,E15.8,25H            RMS VELOCITY=, MATS057
     2E15,8)                                                            MATS058
  20  CONTINUE                                                          MATS059
C     NUMBERS 11-16 RESERVED FOR FUTURE MEASUREMENTS                    MATS060
      IGUID=IGUID                                                       MATS061
      GO TO (18,17),IGUID                                               MATS062
C     IGUID IS SET BY INPUT TO MAIN PROGRAM IF GUIDANCE CALCULATIONS AREMATS063
C     MADE                                                              MATS064
   17 CONTINUE                                                          MATS065
      CALL MATRX(AO,PFTA,PFTA,6,6,6,1)                                  MATS065
      DO 31 I=1,6                                                       MATS066
      DO 31 J=I,6                                                       MATS066
      PFTA(I,J)=(PFTA(I,J)+PFTA(J,I))/2.                                MATS067
      PFTA(J,I)=PFTA(I,J)                                               MATS067
  31  CONTINUE                                                          MATS068
      CALL INVAO(AO,AOI)                                                MATS068
      CALL MATRX(TFTA,AOI,TFTA,3,6,6,0)                                 MATS069
      CALL GUID                                                         MATS071
C     GUID PERFORMS THE GUIDANCE CALCULATIONS                           MATS072
   18 CONTINUE                                                          MATS073
      TSECP=TSEC                                                        MATS074
      RETURN                                                            MATS075
      END                                                               MATS
```

Page Intentionally Left Blank

Subroutine: MNA

Purpose: To provide the rotation matrix EMN which transforms moon centered coordinates in the earth's true equator and equinox to moon centered coordinates in the moon's true equator.

Calling Sequence:

CALL MNA (TIME, OM, CR, DT, EPSIL, RO, G, GP, WW, EMN)

Input and **Output**

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	TIME	1	d	days	Days since 1950.0
I	OM	1	Ω		Output from NUTAIT: (Arg. Decending node)
I	CR	1	☾	radians	Output from NATAIT: (mean long. of the moon)
I	DT	1	$\Delta\Psi + d\Psi$	radians	Output from NUTAIT: (nutation in longitude)
I	EPSIL	1	$\bar{\epsilon} + \Delta\epsilon + d\epsilon$	radians	Output from NUTAIT: (mean obliquity and nutation in obliquity)

(Cont'd.)

Common storages used or required: None.

Subroutines required: System double precision routines.

Functions required: INTF, SQRT, SIN, COS, ACOS

Approximate number of storages required: ______

(Cont'd.)

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
0	RO	1	ρ		Libration term
0	G	1	g	radians	Mean anomaly of the moon
0	GP	1	g	radians	Mean anomaly of the sun
0	WW	1	ω	radians	Argument of the perigee of the
					moon
0	EMN	3,3			Rotation matrix

Transformation From Earth's True Equator to Moon's True Equator.

The two rectangular systems are related through Λ, Ω', and i by the rotation:

$$\underset{\text{MOON}}{\begin{pmatrix} x \\ y \\ z \end{pmatrix}} = \begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{pmatrix} \underset{\text{EARTH}}{\begin{pmatrix} X' \\ Y' \\ Z' \end{pmatrix}}$$

where

$$b_{11} = \cos \Lambda \cos \Omega' - \sin \Lambda \sin \Omega' \cos i$$

$$b_{12} = \cos \Lambda \sin \Omega' + \sin \Lambda \cos \Omega' \cos i$$

$$b_{13} = \cos \Lambda \sin i$$

$$b_{21} = -\sin \Lambda \cos \Omega' - \cos \Lambda \sin \Omega' \cos i$$

$$b_{22} = -\sin \Lambda \sin \Omega' + \cos \Lambda \cos \Omega' \cos i$$

$$b_{23} = \cos \Lambda \sin i$$

$$b_{31} = \sin \Omega' \sin i$$

$$b_{32} = -\cos \Omega' \sin i$$

$$b_{33} = \cos i$$

i is the inclination of the moon's true equator to the earth's equator

Ω' is the right ascension of the ascending node of the moon's true equator

Λ is the anomaly from the node to the X axis

$$\Lambda = \Delta + (☾ + \Upsilon) - (\Omega + \sigma)$$

Δ is the anomaly from the node to the ascending node of the moon's true equator on the ecliptic

Ω is the mean longitude of the descending node of the moon's mean equator on the ecliptic

☾ is the mean longitude of the moon

σ is the libration in the node

Υ is the libration in the mean longitude

ρ is the libration in the inclination.

$\delta\Psi$, ϵ, Ω, and ☾ are input quantities obtained from NUTAIT. The remainder are computed from the following equations.

I = inclination of moon's equator to ecliptic

$I = 1.535^{\circ}$

g = mean anomaly of moon

$g = 215.54013 + 13.064992\ d$

g' = mean anomaly of sun

g' = 358.009067 + .9856005 d

ω = argument of perigee of moon

ω = 196.745632 + .1643586 d

where d = days from 1950.

$$\sigma \sin I = -.0302777 \sin g + .0102777 \sin (g + 2\omega) - .00305555 \sin (2g + 2\omega)$$

$$\tau = -.003333 \sin g + .0163888 \sin g' + .005 \sin 2\omega$$

$$\rho = -.0297222 \cos g + .0102777 \cos (g + 2\omega) - .00305555 \cos (2g + 2\omega)$$

$$\cos i = \cos (\Omega + \sigma + \delta\Psi) \sin \epsilon \sin (I + \rho) + \cos \epsilon \cos (I + \rho) \qquad 0 < i < 90^\circ$$

$$\sin \Omega' = -\sin (\Omega + \sigma + \delta\Psi) \sin (I + \rho) \csc i \qquad -90^\circ < \Omega' < 90^\circ$$

$$\sin \Delta = -\sin (\Omega + \sigma + \delta\Psi) \sin \epsilon \csc i$$

$$\cos \Delta = -\sin (\Omega + \sigma + \delta\Psi) \sin \Omega' \cos \epsilon - \cos (\Omega + \sigma + \delta\Psi) \cos \Omega' \qquad 0^\circ \leq \Delta < 360^\circ$$

Reference: JPL Technical Report No. 32-223

```
*       LABEL                                                          SMNA
*       SYMBOL TABLE                                                   SMNA
 CEC20AH                                                               SMNA
        SUBROUTINE MNA(TIME,OM,CR,DT,EPSIL,RO,G,GP,WW,EM)              SMNA000
        DIMENSION EM(3,3),DF(3)                                        SMNA001
        D = TIME                                                       SMNA002
        T = D/36525.                                                   SMNA003
        T2 = T*T                                                       SMNA004
        T3 = T2*T                                                      SMNA005
        A=13.064992                                                    SMNA006
        DO 6 I=1,3                                                     SMNA007
        DD=D                                                           SMNA008
D       DD=DD*(A/360.)                                                 SMNA009
D       DD=DD-INTF(DD)                                                 SMNA010
        DF(I)=DD                                                       SMNA011
        GO TO (4,5,6),I                                                SMNA012
    4   A=.9856005                                                     SMNA013
        GO TO 6                                                        SMNA014
    5   A=.1643586                                                     SMNA015
    6   CONTINUE                                                       SMNA016
        G=215.54013+360.*DF(1)                                         SMNA017
        GP=358.009067+360.*DF(2)                                       SMNA018
        WW=196.745632+360.*DF(3)                                       SMNA019
        G = G*.017453296                                               SMNA020
        GP = GP*.017453296                                             SMNA021
        WW = WW*.017453296                                             SMNA022
        YN = 1.535*.017453296                                          SMNA023
        RO = -.0297222*COSF(G) + .01020777*COSF(G+2.*WW)               SMNA024
       1 -.00305555*COSF(2.*G+2.*WW)                                   SMNA025
        TA = -.003333*SINF(G) + .0163888*SINF(GP)                      SMNA026
       1 +.005*SINF(2.*WW)                                             SMNA027
        SG = -.0302777*SINF(G) + .0102777*SINF(G+2.*WW)                SMNA028
       1 -.00305555*SINF(2.*G+2.*WW)                                   SMNA029
        SG = (SG*.017453296)/SINF(YN)                                  SMNA030
        RO = RO*.017453296                                             SMNA031
        TA = TA*.017453296                                             SMNA032
        YN = YN + RO                                                   SMNA033
        RO = OM + SG + DT                                              SMNA034
        CI = COSF(RO)*SINF(EPSIL)*SINF(YN)                             SMNA035
       1 +COSF(EPSIL)*COSF(YN)                                         SMNA036
        SI = 1. - CI**2                                                SMNA037
        SI = SQRTF(SI)                                                 SMNA038
        SO = -SINF(RO)*SINF(YN)/SI                                     SMNA039
        CO = 1. - SO**2                                                SMNA040
        CO = SQRTF(CO)                                                 SMNA041
        SD = -SINF(RO)*SINF(EPSIL)/SI                                  SMNA042
        CD = -SINF(RO)*SO*COSF(EPSIL) - COSF(RO)*CO                    SMNA043
        DL = ACOSF(CD)                                                 SMNA044
        IF(SD)1,3,3                                                    SMNA045
  1     DL = 6.283185306 - DL                                          SMNA046
  3     CONTINUE                                                       SMNA047
```

MNA (Cont'd)

```
CA = DL + (CR + TA) - (OM + SG)                    SMNA048
SA = SINF(CA)                                      SMNA049
CA = COSF(CA)                                      SMNA050
RO = COSF(RO)*SINF(EPSIL)/(SI*CD)                  SMNA051
EM(1,1) = CA*CO - SA*SO*CI                         SMNA052
EM(1,2) = CA*SO + SA*CO*CI                         SMNA053
EM(1,3) = SA*SI                                    SMNA054
EM(2,1) = -SA*CO - CA*SO*CI                        SMNA055
EM(2,2) = -SA*SO + CA*CO*CI                        SMNA056
EM(2,3) = CA*SI                                    SMNA057
EM(3,1) = SO*SI                                    SMNA058
EM(3,2) = -CO*SI                                   SMNA059
EM(3,3) = CI                                       SMNA060
RETURN                                             SMNA061
END                                                SMNA
```

Subroutine: MNAND

Purpose: To compute the matrix M used to perform the velocity transformation corresponding to the position transformation described in MNA.

Calling Sequence:

CALL MNAND (TIME, RO, G, GP, WW, EM, DM)

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	TIME	1	d	days	Days since 1950.0
I	RO	1	ρ	radians	Output from MNA; see description
I	G	1	g	radians	Output from MNA; see description
I	GP	1	g'	radians	Output from MNA; see description
I	WW	1	ω	radians	Output from MNA; see description
I	EM	3,3	M		Output from MNA; matrix M
O	DM	3,3	$\dot{M}$		Matrix $\dot{M}$

Common storages used or required: None

Subroutines required: None

Functions required: COS

Approximate number of storages required:

Theory and Equations

In transforming lunacentric position coordinates relative to the earth's equator and equinox of 1950.0 to coordinates relative to the moon's true equator, three matrices are computed. Matrix A (from ROTEQ) rotates the coordinates to equinox of date, matrix N (from NUTAIT) accounts for the nutation of the earth about its precessing mean equator, and matrix M (from MNA) transforms to the position relative to the moon's true equator.

Then
$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix}_{\text{moon}} = MNA \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}_{\text{earth, 1950.0}}$$

Assuming that $\dot{N} = \dot{A} = 0$

$$\begin{pmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{pmatrix}_{\text{moon}} = MNA \begin{pmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{pmatrix}_{\text{earth, 1950}} + \dot{M}NA \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}_{\text{earth, 1950}}$$

The $\dot{M}$ matrix is computed from the equations:

$$\dot{M} = \begin{pmatrix} M_{21}\dot{\Lambda} & M_{22}\dot{\Lambda} & M_{23}\dot{\Lambda} \\ -M_{11}\dot{\Lambda} & -M_{12}\dot{\Lambda} & -M_{13}\dot{\Lambda} \\ 0 & 0 & 0 \end{pmatrix}$$

where M_{ij} are the elements of M, and

$$\dot{\Lambda} = \dot{\Delta} + \dot{☾} + \dot{t} - \dot{\Omega} - \dot{\sigma}$$

$$\dot{\Delta} = -\rho\ (\dot{\Omega} + \dot{\sigma})$$

$$\dot{☾} = 0.266170762 \times 10^{-5} - 0.12499171 \times 10^{-13}\ T\ \text{rad/sec};\ T = d/36525$$

$$\dot{\Omega} = -0.1069698435 \times 10^{-7} + 0.23015329 \times 10^{-3}\ T\ \text{rad/sec}$$

$$\dot{t} = -0.1535272946 \times 10^{-9} \cos g$$
$$+0.569494067 \times 10^{-10} \cos g$$
$$+0.579473484 \times 10^{-11} \cos 2\omega\ \ \text{rad/sec}$$

$$\dot{\sigma} = -0.520642191 \times 10^{-7} \cos g$$
$$+0.1811774451 \times 10^{-7} \cos (g+2\omega)$$
$$-0.1064057858 \times 10^{-7} \cos (2\omega+2g)\ \text{rad/sec}$$

Reference: JPL Technical Report No. 32-223

MNAND

```
*      LABEL                                                              MNAN
*      SYMBOL TABLE                                                       MNAN
       SUBROUTINE MNAND(TIME,RO,G,GP,WW,EM,DM)                            MNAN0000
       DIMENSION DM(3,3),EM(3,3)                                          MNAN0010
       T = TIME/36525.                                                    MNAN0020
       CRD = .266170762E-5 - .12499171E-13*T                              MNAN0030
       OMD = -.1069698435E-7 + .23015329E-13*T                            MNAN0040
       TAD = -1.535272946*COSF(G) + .569494067*COSF(GP)                   MNAN0050
      1 + .0579473484*COSF(2.*WW)                                         MNAN0060
       TAD = .1E-9*TAD                                                    MNAN0070
       SGD = -.520642191*COSF(G) + .1811774451*COSF(G+2.*WW)              MNAN0080
      1 -.1064057858*COSF(2.*WW+2.*G)                                     MNAN0090
       SGD = .1E-6*SGD                                                    MNAN0100
       DLD = -RO*(OMD + SGD)                                              MNAN0110
       CAD = DLD + CRD + TAD - OMD - SGD                                  MNAN0120
       DM(1,1) = EM(2,1)*CAD                                              MNAN0130
       DM(1,2) = EM(2,2)*CAD                                              MNAN0140
       DM(1,3) = EM(2,3)*CAD                                              MNAN0150
       DM(2,1) = -EM(1,1)*CAD                                             MNAN0160
       DM(2,2) = -EM(1,2)*CAD                                             MNAN0170
       DM(2,3) = -EM(1,3)*CAD                                             MNAN0180
       DM(3,1) = 0.0                                                      MNAN0190
       DM(3,2) = 0.0                                                      MNAN0200
       DM(3,3) = 0.0                                                      MNAN0210
       RETURN                                                             MNAN0220
       END                                                                MNAN
```

Subroutine: MONBTR

Purpose: To obtain the position and velocity vectors of the vehicle relative to the beacon and determine if the beacon is in view of the vehicle. The station location partials are computed and subroutine COMPHQ called to update the state covariance matrix for the measurements being made. The types of measurements possible are range, range rate, right ascension, and declination.

Calling Sequence:

CALL MONBTR

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I					Quantities obtained from common
O					Quantities placed in common

Common storages used or required: T, S, C, IC

Subroutines required: COMPHO, DOT, INTRI, MATRX, MNAND, MNA, NUTAIT, TRAC

Functions required: ASIN, ATAN, FNORM, SQRT (FIL)(SLO)(STH)

Approximate number of storages required: 495 DEC

MONBTR Partial Derivatives

MONBTR computes two partial matrices which are used in COMPHQ to evaluate station location errors. The matrices are the following:

$$\text{STPARS} = \begin{pmatrix} \frac{\partial x_1}{\partial LAT} & \frac{\partial x_1}{\partial LON} & \frac{\partial x_1}{\partial ALT} \\ \frac{\partial x_1}{\partial LAT} & \frac{\partial x_2}{\partial LON} & \frac{\partial x_2}{\partial ALT} \\ \frac{\partial x_3}{\partial LAT} & \frac{\partial x_3}{\partial LON} & \frac{\partial x_3}{\partial ALT} \end{pmatrix}.$$

$$\text{STPARD} = \begin{pmatrix} \frac{\partial \dot{x}_1}{\partial LAT} & \frac{\partial \dot{x}_1}{\partial LON} & \frac{\partial \dot{x}_1}{\partial ALT} \\ \frac{\partial \dot{x}_2}{\partial LAT} & \frac{\partial \dot{x}_2}{\partial LON} & \frac{\partial \dot{x}_2}{\partial ALT} \\ \frac{\partial \dot{x}_3}{\partial LAT} & \frac{\partial \dot{x}_3}{\partial LON} & \frac{\partial \dot{x}_3}{\partial ALT} \end{pmatrix}$$

where $\vec{X} = \begin{pmatrix} X_1 \\ X_2 \\ X_3 \end{pmatrix} = -\vec{X}_V + \vec{X}_B \quad \equiv$ XREL (FORTRAN)

$\dot{\vec{X}} = \begin{pmatrix} \dot{X}_1 \\ \dot{X}_2 \\ \dot{X}_3 \end{pmatrix} = -\dot{\vec{X}}_v + \dot{\vec{X}}_B \quad \equiv$ VREL (FORTRAN)

$$\frac{\partial x_i}{\partial LAT} = \frac{\partial x_{Bi}}{\partial LAT} \qquad \frac{\partial \dot{x}_i}{\partial LAT} = \frac{\partial \dot{x}_{Bi}}{\partial LAT} \qquad i = 1,2,3$$

$$\vec{X}_B = (MNA)^T \; \vec{X}_{BM}$$

$$\dot{\vec{X}}_B = (MNAND)^T \; \vec{X}_{BM} + \cancel{(MNA)^T \dot{\vec{X}}_{BM}}^{(=0)}$$

where subscripts BM indicate beacon moon-fixed coordinates.

The transformations MNA and MNAND are obtained by calling the subroutines MNA and MNAND.

$$\underset{3 \times 1}{\vec{X}_{BM}} = \underset{3 \times 3}{(R + ALT)} \underset{3 \times 1}{\begin{pmatrix} \cos LAT \cos LON \\ \cos LAT \sin LON \\ \sin LAT \end{pmatrix}}$$

$$\begin{pmatrix} \frac{\partial X_1}{\partial PAR} \\ \frac{\partial X_2}{\partial PAR} \\ \frac{\partial X_3}{\partial PAR} \end{pmatrix} = (MNA)^T \begin{pmatrix} \frac{\partial X_{BM}}{\partial PAR} \\ \frac{\partial Y_{BM}}{\partial PAR} \\ \frac{\partial Z_{BM}}{\partial PAR} \end{pmatrix}$$

$$\begin{pmatrix} \frac{\partial \dot{X}_1}{\partial PAR} \\ \frac{\partial \dot{X}_2}{\partial PAR} \\ \frac{\partial \dot{X}_3}{\partial PAR} \end{pmatrix} = (MNAND)^T \begin{pmatrix} \frac{\partial X_{BM}}{\partial PAR} \\ \frac{\partial Y_{BM}}{\partial PAR} \\ \frac{\partial Z_{BM}}{\partial PAR} \end{pmatrix}$$

$$\underset{3 \times 1}{\frac{\partial \vec{X}_{BM}}{\partial LAT}} = (R + ALT) \underset{3 \times 1}{\begin{pmatrix} -\sin LAT \cos LON \\ -\sin LAT \sin LON \\ \cos LAT \end{pmatrix}}$$

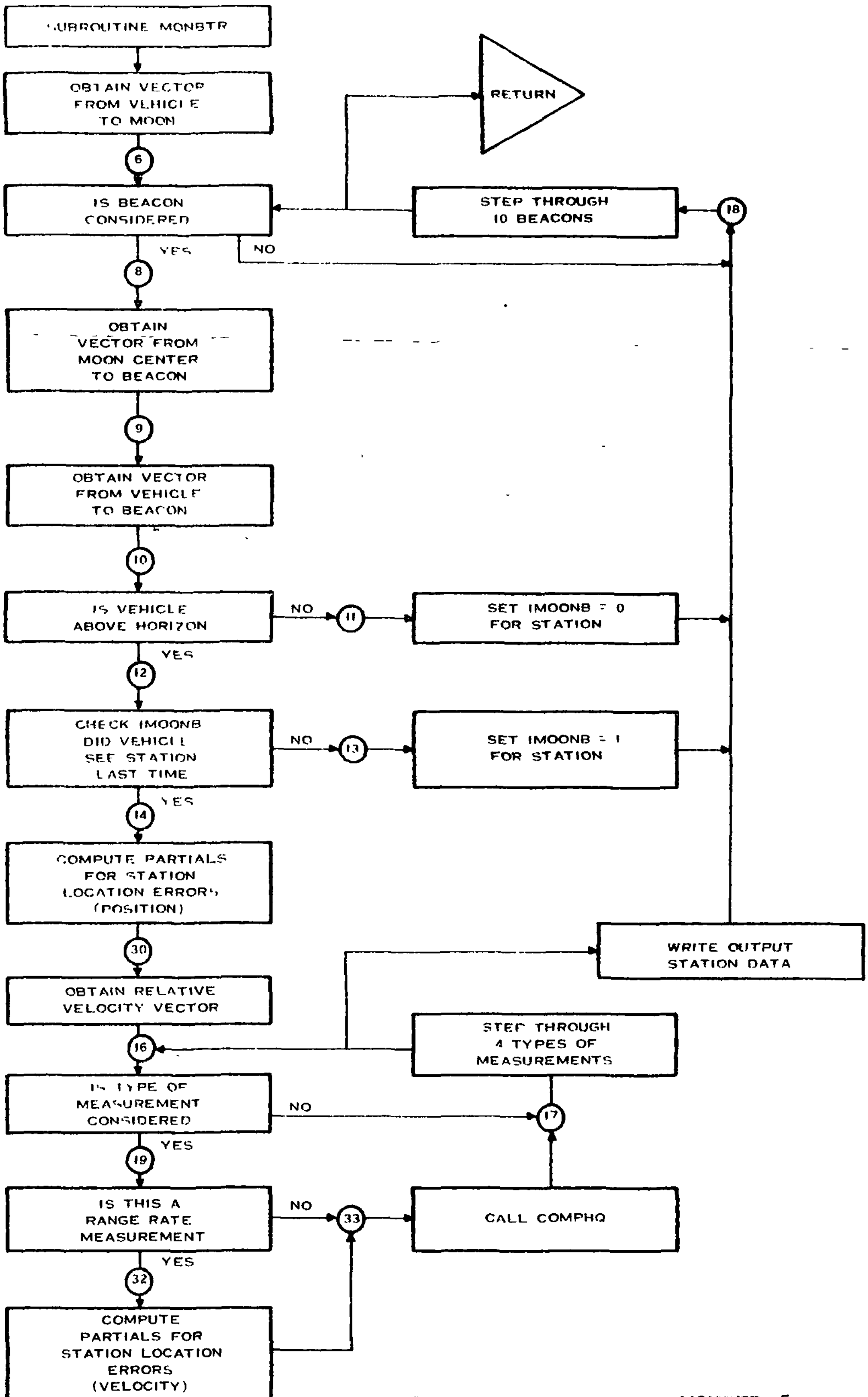
$$\underset{3 \times 1}{\frac{\partial \vec{X}_{BM}}{\partial LON}} = (R + ALT) \underset{3 \times 1}{\begin{pmatrix} -\cos LAT \sin LON \\ \cos LAT \cos LON \\ 0 \end{pmatrix}}$$

$$\underset{3 \times 1}{\frac{\partial \vec{X}_{BM}}{\partial ALT}} = \underset{3 \times 1}{\begin{pmatrix} \cos LAT \cos LON \\ \cos LAT \sin LON \\ \sin LAT \end{pmatrix}}$$

$$\underset{3 \times 3}{STPARS} = \underset{3 \times 3}{(MNA)^T} \underset{3 \times 3}{\left(\frac{\partial \vec{X}_{BM}}{\partial LAT} ; \frac{\partial \vec{X}_{BM}}{\partial LON} ; \frac{\partial \vec{X}_{BM}}{\partial ALT} \right)}$$

$$\underset{3 \times 3}{STPARD} = \underset{3 \times 3}{(MNAND)^T} \underset{3 \times 3}{\left(\frac{\partial \vec{X}_{BM}}{\partial LAT} ; \frac{\partial \vec{X}_{BM}}{\partial LON} ; \frac{\partial \vec{X}_{BM}}{\partial ALT} \right)}$$

Following computation of the matricies, subroutine COMPHQ is called to perform the updating of the state covariance matrix.


SUBROUTINE MONBTR
OBTAIN VECTOR FROM VEHICLE TO MOON
RETURN
6
IS BEACON CONSIDERED
STEP THROUGH 10 BEACONS
18
YES
NO
8
OBTAIN VECTOR FROM MOON CENTER TO BEACON
9
OBTAIN VECTOR FROM VEHICLE TO BEACON
10
IS VEHICLE ABOVE HORIZON
NO
11
SET IMOONB = 0 FOR STATION
YES
12
CHECK IMOONB DID VEHICLE SEE STATION LAST TIME
NO
13
SET IMOONB = 1 FOR STATION
YES
14
COMPUTE PARTIALS FOR STATION LOCATION ERRORS (POSITION)
WRITE OUTPUT STATION DATA
30
OBTAIN RELATIVE VELOCITY VECTOR
STEP THROUGH 4 TYPES OF MEASUREMENTS
16
IS TYPE OF MEASUREMENT CONSIDERED
NO
17
YES
19
IS THIS A RANGE RATE MEASUREMENT
NO
33
CALL COMPHQ
YES
32
COMPUTE PARTIALS FOR STATION LOCATION ERRORS (VELOCITY)

MONBTR

```
*       LABEL                                                                   MONB
*       SYMBOL TABLE                                                            MONB
CEC2012  SUBROUTINE MONBTR                                                      MONB
        SUBROUTINE MONBTR                                                       MONB0000
        COMMON T,S,C,IC                                                         MONB0010
        DIMENSION                                                               MONB0020
       1T(1360),S(1000),C(1000),IC(1),XREL(3),VREL(3)                           MONB0030
       1,PO(22),VE(22),AN(3,3),EM(3,3),ISBCON(10),U(3),EN(3)                    MONB0040
       2,RT(3),E(3),ETM(3,3),IMOONB(10),DM(3,3),DUM(3,3)                        MONB0050
       3,RMB(3),TYMEAS(4,10),X(3),VX(3),OUTPUT(6)                               MONB0060
        DIMENSION STPARS(3,3),DUD(3,3),STPARD(3,3)                              MONB0065
        EQUIVALENCE (T,TDUM),(S,SDUM),(C,CDUM),(IC,ICDUM)                       MONB0070
       1,(C(895),XREL),(C(898),VREL),(C(894), OBNO),(C(893),XMAG),              MONB0080
       2(C(892),DEX),(C(891),DEN1),(C(890),DEN2)                                MONB0090
       3,(C(889),DEX2),(C(888),DEX3),(C(652),P),(C(973),OUTPUT)                 MONB0100
       1,(C(13),TW),(C(14),TF),(C(62),PO),(IC(3),NOR)                           MONB0110
       2,(IC(190),LSTAT),(C(138),AN),(S(24),A),(C(120),EM)                      MONB0120
       3,(IC(140),ISBCON),(S(72),DR),(IC(220),IMOONB),(C(649),TSECP)            MONB0130
       4,(C(30),TSEC),(IC(140),TYMEAS),(C(15),X),(C(18),VX)                     MONB0140
        EQUIVALENCE (C(800),STPARD),(C(788),STPARS)                             MONB0145
        CALL SETN(NIN,NUTS)                                                     MONB0147
        DELTT=TSEC-TSECP                                                        MONB0150
        LSTAT=LSTAT                                                             MONB0160
        GO TO(1,2),LSTAT                                                        MONB0170
  1     CONTINUE                                                                MONB0180
        NB=NOR-1                                                                MONB0190
        DIS=1.E10                                                               MONB0200
        CALL INTR1(TW,TF,NB,PO,1,VE,DIS)                                        MONB0210
  2     CONTINUE                                                                MONB0220
        TIME=TW+TF                                                              MONB0230
        CALL NUTAIT(TIME,OM,CR,DT,EM,EPSIL)                                     MONB0240
        CALL MNA(TIME,OM,CR,DT,EPSIL,RO,G,GP,WW,ETM)                            MONB0250
        CALL MNAND(TIME,RO,G,GP,WW,ETM,DM)                                      MONB0260
        IF(NOR-2)3,5,3                                                          MONB0270
  3     CONTINUE                                                                MONB0280
        DO 4 I=1,3                                                              MONB0290
        J=20-I                                                                  MONB0300
        XREL(I)=PO(J)-X(I)                                                      MONB0310
        VREL(I)=VE(J)-VX(I)                                                     MONB0320
  4     CONTINUE                                                                MONB0330
        GO TO 7                                                                 MONB0340
  5     CONTINUE                                                                MONB0350
        DO 6 I=1,3                                                              MONB0360
        XREL(I)=-X(I)                                                           MONB0370
        VREL(I)=-VX(I)                                                          MONB0380
  6     CONTINUE                                                                MONB0390
C       XREL POSITION VECTOR FROM VEHICLE TO MOON CENTER 1950                   MONB0400
C       VREL RELATIVE VELOCITY OF MOON CENTER 1950                              MONB0410
  7     CONTINUE                                                                MONB0420
        NN=485                                                                  MONB0430
        DO 18 III=1,10                                                          MONB0440
```

INTR, INTRI(Cont'd)

```
        SUB     TARGO
        TNZ     NEU                 TIME HAS CHANGED
        CLA     TARG+1
        SUB     TARGO+1
        TZE     FLEE+6
        REM                         ANALYZE TABLE
        REM                         NEEDS AS A
        REM                         FUNCTION OF
        REM                         CENTRAL BODY
NEU     CLA     TARG
        FSB     TABLE
        TMI     LOOKUP              POSITION EPHEMERIS TAPE
        STO     COM+19              T-TO
        SUB     20.
        TPL     LOOKUP
        CLA     COM+19
        FDH     4.
        STQ     COM+18
        CLA     COM+18              (T-TO)/4.
        CALL    FIX
        CALL    FLOAT
        STO     COM+17
        CHS
        FAD     COM+18
        STO     COM+1
        CLA     TARG+1
        FDH     4.
        STQ     COM
        CLA     COM
        FAD     COM+1
        STO     TARG+2
        LDQ     COM+19
        FMP     =9.
        CALL    FIX
        ADD     KERNO+1
        STA     GG
        LDQ     COM+17
        FMP     =54.
        CALL    FIX
        ADD     KERNO+2
        STA     HH
        AXT     2*SEP,4             MUST INTERPOLATE
        STZ     XN+2*SEP,4          SO CLEAR STORAGE
        TIX     *-1,4,1
        AXT     BSEP,4              SET GRAVITATIONAL
        STZ     KBO+BSEP,4          COEFFICIENTS
        TIX     *-1,4,1             TO ZERO
        LXA     CENTER,4
        PXD     ,4
        COM
        PDX     ,2
```

INTR, INTRI (Cont'd)

```
         TXH      HELIO,4,1                                              INTR1010
GEO      AXT      3,1                                                    INTR1020
         CLA      GRAV+3,1                                               INTR1030
         STO      KBO+3,1                                                INTR1040
         TIX      *-2,1,1                                                INTR1050
         STZ      KBO-1,2                                                INTR1060
         CLA      EWORTO                                                 INTR1070
         STO      EWORT                                                  INTR1080
         CLA      HWORTE                                                 INTR1090
         STO      HWORT                                                  INTR1100
         TXH      GG12,4,0                                               INTR1110
         CLA      RJ                  TEST FOR                           INTR1120
         SUB      R                   INCLUSION OF                       INTR1130
         TPL      GG12                JUPITER IF                         INTR1140
         CLA      HWORTJ              EARTH IS                           INTR1150
         STO      HWORT               CENTRAL BODY                       INTR1160
         CLA      GRAV+6                                                 INTR1170
         STO      KB6                                                    INTR1180
         TXL      GG12,,**                                               INTR1190
HELIO    AXT      BSEP,1                                                 INTR1200
         CLA      GRAV+BSEP,1                                            INTR1210
         STO      KBO+BSEP,1                                             INTR1220
         TIX      *-2,1,1                                                INTR1230
         STZ      KB2-3,2                                                INTR1240
ORTHO    CLA      EWORTO                                                 INTR1250
         STO      EWORT                                                  INTR1260
         CLA      HWORTO                                                 INTR1270
         STO      HWORT                                                  INTR1280
GG12     CLA      VEL                 CHECK                              INTR1290
         TZE      GG6                 FOR VELOCITY                       INTR1300
         CLA      EWORTO              OPTIONS                            INTR1310
         STO      EWORT                                                  INTR1320
         CLA      HWORTV                                                 INTR1330
         STO      HWORT                                                  INTR1340
GG6      CLA      EWORT                                                  INTR1350
         TZE      GG+2                                                   INTR1360
         CLA      EVEL                                                   INTR1370
         STO      VEL+1                                                  INTR1380
         CLA      TARG+1              GEOCENTRIC                         INTR1390
         TSX      TAB,4               INTERPOLATION                      INTR1400
GG       PZE      **,,9                                                  INTR1410
         PZE      XN,,EWORT+1                                            INTR1420
         CLA      HWORT                                                  INTR1430
         TZE      HH+2                                                   INTR1440
         CLA      HVEL                                                   INTR1450
         STO      VEL+1                                                  INTR1460
         CLA      TARG+2              HELIOCENTRIC                       INTR1470
         TSX      TAB,4               INTERPOLATION                      INTR1480
HH       PZE      **,,54                                                 INTR1490
         PZE      XN+3,,HWORT+1                                          INTR1500
RESET    AXT      3,4                 PLACE                              INTR1510
```

INTR, INTRI (Cont'd)

```
        CLA     XN+6,4            COORDINATES                        INTR152
        FDH     EVEL              IN OLD                             INTR153
        STQ     XN+18,4           FORMAT                             INTR154
        CLA     XN+9,4                                               INTR155
        FDH     EVEL                                                 INTR156
        STQ     XN+12,4                                              INTR157
        LDQ     XN+3,4                                               INTR158
        STQ     XN+6,4                                               INTR159
        FMP     MU                                                   INTR160
        FSB     XN+18,4                                              INTR161
        STO     XN+9,4            POSITION OF SUN                    INTR162
        STZ     XN+3,4                                               INTR163
        LDQ     XN+3,4                                               INTR164
        STQ     XN+6,4                                               INTR165
        FMP     MU                                                   INTR166
        FSB     XN+18,4                                              INTR167
        STO     XN+9,4            VELOCITY OF SUN                    INTR168
        STZ     XN+3,4                                               INTR169
        TIX     RESET+1,4,1                                          INTR170
        LXA     CENTER,4                                             INTR171
        LXD     VEL,2             IX2 NOT ZERO IF VELOCITY OPTION    INTR172
        TXH     RVPRT,2,0         GET COORDINATES FOR PRINTING       INTR173
        TXH     TESTM,4,0                                            INTR174
        NZT     KB6               EARTH-CENTERED                     INTR175
        TRA     FLEE                                                 INTR176
        AXT     3,1                                                  INTR177
        CLA     XN+21,1                                              INTR178
        FAD     XN+9,1                                               INTR179
        STO     XN+21,1                                              INTR180
        TIX     *-3,1,1                                              INTR181
        TXL     FLEE                                                 INTR182
TESTM   TXH     RVPRT,4,1                                            INTR183
        AXT     3,1               MOON-CENTERED                      INTR184
        CLS     XN+6,1                                               INTR185
        STO     XN+3,1                                               INTR186
        CLA     XN+9,1                                               INTR187
        FSB     XN+6,1                                               INTR188
        STO     XN+9,1                                               INTR189
        STZ     XN+6,1                                               INTR190
        TIX     *-6,1,1                                              INTR191
FLEE    CLA     CENTER                                               INTR192
        STO     KERNO                                                INTR193
        CLA     TARG                                                 INTR194
        STO     TARGO                                                INTR195
        CLA     TARG+1                                               INTR196
        STO     TARGO+1                                              INTR197
        AXT     21,1              TRANSFER POSITIONS                 INTR198
        CLA     XN+21,1                                              INTR199
OUP     STO     **,1                                                 INTR200
        TIX     OUP-1,1,1                                            INTR201
        CLA     VEL                                                  INTR202
```

INTR, INTRI (Cont'd)

```
        TZE     OUT                      DO NOT TRANSFER VELOCITY          INTR2030
        AXT     21,1                                                        INTR2040
        CLA     XN,+21,1                 TRANSFER VELOCITIES                INTR2050
(N,V    STO     **,1                                                        INTR2060
        TIX     XN,V-1,1,1                                                  INTR2070
)UT     LXD     HELIO-1,1                                                   INTR2080
        LXD     TRAP+1,2                                                    INTR2090
        LXD     TRAP,4                                                      INTR2100
        TRA     8,4                                                         INTR2110
        REM     TSX TAB,4                (AC)=INTERPOLATIVE ARGUMENT        INTR2120
        REM     PZE B,,K                                                    INTR2130
        REM     PZE A,,C                                                    INTR2140
        REM                              B=START OF DATA BLOCK              INTR2150
        REM                              K=WORDS PER SUB BLOCK              INTR2160
        REM                              A=START OF RESULT BLOCK            INTR2170
        REM                              C=SKIP CODE WORD LOCATION          INTR2180
'AB     SXD     COM+9,4                                                     INTR2190
        SXD     COM+8,2                                                     INTR2200
        SXD     COM+7,1                                                     INTR2210
        STO     ARG                                                         INTR2220
        CLA     1,4                                                         INTR2230
        STA     TAB18                                                       INTR2240
        LRS     18                                                          INTR2250
        ADD     1,4                                                         INTR2260
        STA     TAB21                                                       INTR2270
        CLA     2,4                                                         INTR2280
        PAX     ,1                                                          INTR2290
        TXI     *+1,1,SEP                                                   INTR2300
        SXA     TAB29,1                                                     INTR2310
        ARS     18                                                          INTR2320
        STA     VELOP                                                       INTR2330
        AXT     2,1                                                         INTR2340
        CLA     VEL                                                         INTR2350
        TNZ     VELOP                                                       INTR2360
        TXI     VELOP,1,-1                                                  INTR2370
ELOP    CLA     **,1                     PICK UP SKIP                       INTR2380
        STO     VEL+2                    CODE WORD                          INTR2390
        TXL     POSOP,1,1                                                   INTR2400
        REM                              VELOCITY OPTION                    INTR2410
        CLA     ARG                                                         INTR2420
        TSX     COEFF.,4                 FORM THE                           INTR2430
        AXT     3,4                      E1(2J).                            INTR2440
        CLS     COM+13,4                                                    INTR2450
        STO     COM+16,4                                                    INTR2460
        TIX     *-2,4,1                                                     INTR2470
        CLA     1.                       FORM THE                           INTR2480
        FSB     ARG                      E0(2J).                            INTR2490
        TSX     COEFF.,4                                                    INTR2500
        TRA     TAB1-1                                                      INTR2510
        REM                              POSITION OPTION                    INTR2520
OSOP    CLA     ARG                                                         INTR2530
```

INTR, INTRI (Cont'd)

```
         TSX    COEFF,4               FORM THE           INTR254
         AXT    3,4                   E1(2J)             INTR255
         CLA    COM+13,4                                 INTR256
         STO    COM+16,4                                 INTR257
         TIX    *-2,4,1                                  INTR258
         CLA    1.                    FORM THE           INTR259
         FSB    ARG                   E0(2J)             INTR260
         TSX    COEFF,4                                  INTR261
         LXA    TAB29,4                                  INTR262
         TXI    *+1,4,-SEP                               INTR263
         SXA    TAB29,4                                  INTR264
         SXD    COM+4,1                                  INTR265
TAB1     AXT    0,3                                      INTR266
         LDQ    VEL+2                                    INTR267
         PXD                                             INTR268
         LGL    2                                        INTR269
         PAX    ,4                                       INTR270
         STQ    VEL+2                                    INTR271
         CAL    =03                                      INTR272
         ORS    VEL+2                                    INTR273
         TXH    END,4,2                                  INTR274
         TXH    FORT,4,0                                 INTR275
         TXI    *+1,1,-3                                 INTR276
         TXI    TAB1+1,2,-1                              INTR277
FORT     AXT    3,4                                      INTR278
TAB18    LDQ    **,1                  X(0)               INTR279
         FMP    COM+13,4                                 INTR280
         STO    COM                                      INTR281
TAB21    LDQ    **,1                  X(1)               INTR282
         FMP    COM+16,4                                 INTR283
         FAD    COM                                      INTR284
         STO    COM+4,4                                  INTR285
         TXI    *+1,1,-1                                 INTR286
         TIX    *-8,4,1                                  INTR287
         FAD    COM+2                                    INTR288
         FAD    COM+1                                    INTR289
TAB29    STO    **,2                  X(T)               INTR290
         TXI    TAB1+1,2,-1                              INTR291
END      LXD    COM+4,1                                  INTR292
         TIX    VELOP,1,1                                INTR293
         LXD    COM+9,4                                  INTR294
         LXD    COM+8,2                                  INTR295
         LXD    COM+7,1                                  INTR296
         TRA    3,4                                      INTR297
COEFF    STO    COM+10                CALCULATE          INTR298
         LDQ    COM+10                POSITION           INTR299
         FMP    COM+10                COEFFICIENTS       INTR300
         STO    COM+12                FOR EVERETT,S      INTR301
         FSB    1.                    INTERPOLATION      INTR302
         FDH    6.                    FORMULA            INTR303
         FMP    COM+10                                   INTR304
```

INTR, INTRI (Cont'd)

```
        STO    COM+11                                    INTR3050
        CLA    COM+12                                    INTR3060
        FSB    4.                                        INTR3070
        FDH    20.                                       INTR3080
        FMP    COM+11                                    INTR3090
        STO    COM+12                                    INTR3100
        TRA    1,4                                       INTR3110
COEFF.  STO    COM              CALCULATE                INTR3120
        CLS    1.               VELOCITY                 INTR3130
        FDH    VEL+1            COEFFICIENTS             INTR3140
        STQ    COM+10           FOR EVERETT,S            INTR3150
        LDQ    COM              INTERPOLATION            INTR3160
        FMP    COM              FORMULA                  INTR3170
        STO    COM+12                                    INTR3180
        XCA                                              INTR3190
        FMP    3.                                        INTR3200
        FSB    1.                                        INTR3210
        FDH    6.                                        INTR3220
        FMP    COM+10                                    INTR3230
        STO    COM+11                                    INTR3240
        CLA    COM+12                                    INTR3250
        FSB    3.                                        INTR3260
        XCA                                              INTR3270
        FMP    5.                                        INTR3280
        XCA                                              INTR3290
        FMP    COM+12                                    INTR3300
        FAD    4.                                        INTR3310
        FDH    120.                                      INTR3320
        FMP    COM+10                                    INTR3330
        STO    COM+12                                    INTR3340
        TRA    1,4                                       INTR3350
RVPRT   AXT    0,2              EXPRESS ALL              INTR3360
        AXT    3,5              BODIES GEOCENTRICALLY    INTR3370
        CLA    XN+9,2                                    INTR3380
        FAD    XN+9,4                                    INTR3390
        STO    XN+9,2                                    INTR3400
        CLA    VEL                                       INTR3410
        TZE    *+4                                       INTR3420
        CLA    XN.+9,2                                   INTR3430
        FAD    XN.+9,4                                   INTR3440
        STO    XN.+9,2                                   INTR3450
        TXI    *+1,2,-1                                  INTR3460
        TIX    RVPRT+2,4,1                               INTR3470
        TXH    RVPRT+1,2,9-SEP                           INTR3480
        CLA    CENTER           BUFFER                   INTR3490
        ALS    1                CENTRAL                  INTR3500
        ADD    CENTER           BODY                     INTR3510
        PAC    ,4                                        INTR3520
        CLA    XN,4                                      INTR3530
        STO    COM+3,1                                   INTR3540
        CLA    XN.,4                                     INTR3550
```

INTR, INTRI (Cont'd)

```
        STO     COM+6,1                                        INTR3560
        TXI     *+1,4,-1                                       INTR3570
        TIX     *-5,1,1                                        INTR3580
RVPRT1  AXT     0,2                   EXPRESS ALL              INTR3590
        AXT     3,4                   BODIES IN TERMS          INTR3600
        CLA     XN,2                  OF THE CENTRAL           INTR3610
        FSB     COM+3,4               BODY                     INTR3620
        STO     XN,2                                           INTR3630
        CLA     VEL                                            INTR3640
        TZE     *+4                                            INTR3650
        CLA     XN,,2                                          INTR3660
        FSB     COM+6,4                                        INTR3670
        STO     XN,,2                                          INTR3680
        TXI     *+1,2,-1                                       INTR3690
        TIX     RVPRT1+2,4,1                                   INTR3700
        TXH     RVPRT1+1,2,-SEP                                INTR3710
        TXL     FLEE                                           INTR3720
LOOKUP  CLA     TLAST                                          INTR3730
        FSB     TARG                                           INTR3740
        TNZ     *+2                                            INTR3750
        TRA     ERR1                                           INTR3760
        TMI     *-1                                            INTR3770
        CLA     TARG                                           INTR3780
        FSB     TFIRST                                         INTR3790
        TPL     *+2                                            INTR3800
        TRA     ERR1                                           INTR3810
        FDH     20.                                            INTR3820
        XCA                                                    INTR3830
        CALL    FIX                                            INTR3840
        CALL    FLOAT                                          INTR3850
        XCA                                                    INTR3860
        FMP     20.                                            INTR3870
        FAD     TFIRST                                         INTR3880
        STO     COM+1                 TIME ON RECORD           INTR3890
        CLA     TABLE                                          INTR3900
        FSB     COM+1                                          INTR3910
        TMI     FINDIT                                         INTR3920
        FDH     20.                                            INTR3930
        XCA                                                    INTR3940
        CALL    FIX                                            INTR3950
        ADD     1F                    RECORDS TO               INTR3960
        PAX     ,4                    BE BACKED                INTR3970
        TEFA    *+1                                            INTR3980
        TXL     *+3,4,20              OVER                     INTR3990
        REWA    6                                              INTR4000
        TRA     FINDIT                                         INTR4010
        BSRA    6                                              INTR4020
        TIX     *-1,4,1                                        INTR4030
FINDIT  AXT     10,2                                           INTR4040
        SDHA    6                                              INTR4050
        RTBA    6                                              INTR4060
```

INTR, INTRI (Cont'd)

```
         RCHA    IO                                                     INTR4070
         TCOA    *                                                      INTR4080
         TEFA    ERR1                                                   INTR4090
         CLA     TABLE                                                  INTR4100
         SUB     COM+1                                                  INTR4110
         TMI     *+2                                                    INTR4120
         TNZ     LOOKUP              TAPE NOT POSITIONED PROPERLY       INTR4130
         TNZ     FINDIT                                                 INTR4140
         AXT     513,4                                                  INTR4150
         CAL     TABLE               CHECK                              INTR4160
         ACL     A+513,4             SUM                                INTR4170
         TIX     *-1,4,1                                                INTR4180
         TRCA    BSRA                                                   INTR4190
         LAS     C                                                      INTR4200
         TRA     *+2                                                    INTR4210
         TRA     SCALE                                                  INTR4220
BSRA     BSRA    6                                                      INTR4230
         TIX     FINDIT+1,2,1                                           INTR4240
         CALL    ERP                                                    INTR4250
IO       IOCD    TABLE,,515                                             INTR4260
ERR1     CALL    ERPT                                                   INTR4270
         PZE     TARG                                                   INTR4280
         REM                         SCALING                            INTR4290
         REM                         LOOP                               INTR4300
SCALE    AXT     189,4                                                  INTR4310
         LDQ     A+189,4             SCALE                              INTR4320
         FMP     SCALE1              GEOCENTRIC                         INTR4330
         STO     A+189,4             EPHEMERIS                          INTR4340
         TIX     *-3,4,1                                                INTR4350
         AXT     324,4                                                  INTR4360
         LDQ     B+324,4             SCALE                              INTR4370
         FMP     SCALE2              HELIOCENTRIC                       INTR4380
         STO     B+324,4             EPHEMERIS                          INTR4390
         TIX     *-3,4,1                                                INTR4400
         CLA     GRAV                COMPUTE                            INTR4410
         FAD     GRAV+1              MASS RATIO                         INTR4420
         STO     COM                 OF MOON                            INTR4430
         CLA     GRAV+1              TO EARTH-MOON                      INTR4440
         FDH     COM                 BARYCENTER                         INTR4450
         STQ     MU                  FOR POSITION OF EARTH              INTR4460
         TRA     NEU                                                    INTR4470
SCALE1   DEC     6378.150            EARTH RADIUS                       INTR4480
SCALE2   DEC     149598500.          ASTRONOMICAL UNIT (JPL, JULY 1961INTR4490
TFIRST   DEC     3892.0              0 HR AUG 28, 1960 JD = 2437174.5 INTR4500
TLAST    DEC     7292.0              0 HR DEC 19, 1969 JD = 2440554.5 INTR4510
TRAP     PZE                                                            INTR4520
         PZE                                                            INTR4530
RJ       DEC     1E6                 JUPITER TEST DISTANCE              INTR4540
TEMPDT   DEC     34.                 E.T.-U.T.                          INTR4550
         PZE                                                            INTR4560
GRAV     DEC     3.98602E5           EARTH                              INTR4570
```

INTR, INTRI (Cont'd)

```
         DEC     4.8983349E3          MOON                               INTR458
         DEC     1.3252312E11         SUN                                INTR459
         DEC     3.2429889E5          VENUS                              INTR460
         DEC     4.2915518E4          MARS                               INTR461
         DEC     0                    REMOVE BARYCENTER                  INTR462
         DEC     1.2652701E8          JUPITER                            INTR463
EVEL     DEC     86400.                                                  INTR464
HVEL     DEC     345600.                                                 INTR465
MU       PZE                          MASS RATIO OF MOON TO BARYCENTER   INTR466
         OCT     000000520052         MARS, JUPITER VELOCITY             INTR467
HWORT    OCT     0                                                       INTR468
         OCT     527777777777         MOON VELOCITY                      INTR469
EWORT    OCT     0                                                       INTR470
HWORTJ   OCT     000000005252         BARYCENTER, JUPITER                INTR471
HWORTE   OCT     000000005200         BARYCENTER                         INTR472
EWORTO   OCT     527777777777                                            INTR473
HWORTO   OCT     000052525252                                            INTR474
HWORTV   OCT     525252525252                                            INTR475
TARGO    DEC     20,,0                FORMER TIME                        INTR476
KERNO    PZE                          FORMER CENTER                      INTR477
         PZE     A                    GEOCENTRIC REFERENCE               INTR478
         PZE     B                    HELIOCENTRIC REFERENCE             INTR479
ARG      PZE     0                                                       INTR480
1F       DEC     1                                                       INTR481
1.       DEC     1.                                                      INTR482
3.       DEC     3.                                                      INTR483
4.       DEC     4.                                                      INTR484
5.       DEC     5.                                                      INTR485
6.       DEC     6.                                                      INTR486
20.      DEC     20.                                                     INTR487
120.     DEC     120.                                                    INTR488
86400.   DEC     86400.                                                  INTR489
VEL      BSS     3                    VELOCITY OPTION,H,SKIP CODE        INTR490
*                                                                        INTR491
TABLE    DEC     0                    RESERVE                            INTR492
A        BSS     189                  FOR                                INTR493
B        BSS     324                  WORKING                            INTR494
C        BSS     1                    EPHEMERIS                          INTR495
*                                                                        INTR496
TARG     PZE                                                             INTR497
         PZE                                                             INTR498
         PZE                                                             INTR499
CENTER   PZE                                                             INTR500
COM      BSS     21                                                      INTR501
VAFLG    PZE                                                             INTR502
FQFLG    BSS     3                                                       INTR503
KB0      PZE                                                             INTR504
KB1      PZE                                                             INTR505
KB2      PZE                                                             INTR506
KB3      PZE                                                             INTR507
KB4      PZE                                                             INTR508
```

INTR, INTRI (Cont'd)

```
KB5     PZE                                                    INTR5090
KB6     PZE                                                    INTR5100
XN      BSS      21                                            INTR5110
XN.     BSS      21                                            INTR5120
T       PZE                                                    INTR5130
        PZE                                                    INTR5140
R       PZE                                                    INTR5150
BSEP    SYN      7                                             INTR5160
SEP     SYN      XN,-XN                                        INTR5170
        END                                                    INTR
```

Subroutine: INV3

Purpose: To invert a matrix of any dimension up to a 6 by 6.

Calling Sequence:

CALL INV3 (A, N, DETERM)

Input and **Output**

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
IO	A	6,6			A(I) Matrix to be inverted
					A(O) Inverse of input matrix
I	N	1			Dimension of input matrix
O	DETERM	1			Matrix determinant

Common storages used or required: None

Subroutines required: None

Functions required: ABS

Approximate number of storages required:

INV3

```
*       LABEL
C       SUBROUTINE FOR INVERTING SQUARE MATRICES WHICH ARE 6 BY 6 OR LESS
C       (TO INVERT LARGER MATRICES,SAY M BY M, DIMENSION IPIVOT(M),A(M*M),
C       INDEX(M,2),PIVOT(M) AND RECOMPILE)
C       A IS THE SQUARE MATRIX TO BE INVERTED
C       N IS THE SIZE OF A (A IS AN N BY N MATRIX)
C       THE SUBROUTINE RETURNS A INVERSE IN PLACE OF A AND THE DETERMINANT
C       IN DETERM.
        SUBROUTINE INV3(A,N,DETERM)
        DIMENSION IPIVOT(6),A(36) ,INDEX(6,2),PIVOT(6)
        EQUIVALENCE (IROW,JROW), (ICOLUM,JCOLUM),(AMAX,T,SWAP)
        DETERM=1.0
        DO 20  J=1,N
 20     IPIVOT(J)=0
        DO 550  I=1,N
        AMAX=0.0
        DO 105 J=1,N
        IF(IPIVOT(J)-1) 60,105,60
 60     DO 100 K=1,N
        IF(IPIVOT(K)-1) 80,100,740
   80   M = N*(K-1)+J
        IF (ABSF(AMAX)-ABSF(A(M))) 85,100,100
 85     IROW=J
        ICOLUM=K
        AMAX = A(M)
 100    CONTINUE
 105    CONTINUE
        IPIVOT(ICOLUM)=IPIVOT(ICOLUM)+1
        IF(IROW-ICOLUM) 140,260,140
 140    DETERM=-DETERM
        DO 200 L=1,N
        M = N*(L-1)
        M1 = M+ICOLUM
        M = M+IROW
        SWAP = A(M)
        A(M) = A(M1)
  200   A(M1) = SWAP
 260    INDEX(I,1)=IROW
        INDEX(I,2)=ICOLUM
        M = N*(ICOLUM-1)+ICOLUM
        PIVOT(I) = A(M)
        DETERM=DETERM*PIVOT(I)
        A(M) = 1.0
        DO 350 L=1,N
        M = N*(L-1)+ICOLUM
  350   A(M) = A(M)/PIVOT(I)
        DO 550 L1=1,N
        IF(L1-ICOLUM) 400,550,400
  400   M = N*(ICOLUM-1)+L1
        T = A(M)
        A(M) = 0.
```

INV3 (Cont'd)

```
      DO 450  L=1,N
      M = N*(L-1)
      M1 = M+ICOLUM
      M = M+L1
  450 A(M) = A(M)-A(M1)*T
 550  CONTINUE
      DO 710 I=1,N
      L=N+1-I
      IF(INDEX(L,1)-INDEX(L,2)) 630,710,630
 630  JROW=INDEX(L,1)
      JCOLUM=INDEX(L,2)
      M = N*(JROW-1)
      M1 = N*(JCOLUM-1)
      DO 705 K=1,N
      M = M+1
      M1 = M1+1
      SWAP = A(M)
      A(M) = A(M1)
      A(M1) = SWAP
 705  CONTINUE
 710  CONTINUE
 740  RETURN
      END
```

Page Intentionally Left Blank

Subroutine: INVAO

Purpose: To form the inverse of a transition matrix for a linear, conservative dynamic system. If Φ is a partioned matrix such as

$$\Phi = \begin{pmatrix} \Phi_1 & \Phi_2 \\ \Phi_3 & \Phi_4 \end{pmatrix} \quad \text{then INVAO computes } \Phi^{-1} = \begin{pmatrix} \Phi_4^T & -\Phi_2^T \\ -\Phi_3^T & \Phi_1^T \end{pmatrix}$$

Calling Sequence:

CALL INVAO (AO, AOI)

Input and **Output**

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	AO	6,6	$\Phi(t_2,t_1)$		Transition Matrix
O	AOI	6,6	$\Phi^{-1}(t_2,t_1)$		Inverse Transition Matrix

Common storages used or required: None

Subroutines required: None

Functions required: None

Approximate number of storages required:

INVAO

```
*       LABEL                                               INVA
*       SYMBOL TABLE                                        INVA
        SUBROUTINE INVAO(AO,AOI)                            INVA0000
        DIMENSION AO(6,6),AOI(6,6)                          INVA0010
        DO 1 I=1,3                                          INVA0020
        II=I+3                                              INVA0030
        DO 1 J=1,3                                          INVA0040
        JJ=J+3                                              INVA0050
        AOI(I,J)=AO(JJ,II)                                  INVA0060
        AOI(II,J)=-AO(JJ,I)                                 INVA0070
        AOI(II,JJ)=AO(J,I)                                  INVA0080
    1   AOI(I,JJ)=-AO(J,II)                                 INVA0090
        RETURN                                              INVA0100
        END                                                 INVA
```

Subroutine: LOADO

Purpose: To transfer the transition matrix from its storage in the integration package's T block to the matrix in the call list.

Calling Sequence:

CALL LOADO (AO)

Input and **Output**

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
O	AO	(6,6)	$\Phi(t_2 t_1)$		Transition Matrix

Common storages used or required: T

Subroutines required: None

Functions required: None

Approximate number of storages required:

LOADO

```
*       LABEL                                                        LOAD
*       SYMBOL TABLE                                                 LOAD
        SUBROUTINE LOADO(AC)                                         LOAD0000
        COMMON T                                                     LOAD0010
        DIMENSION T(1360),AO(6,6)                                    LOAD0020
        K=6                                                          LOAD0030
        JJ=27                                                        LOAD0040
        DO1 I=1,6                                                    LOAD0050
        DO 1 J=1,3                                                   LOAD0060
        K=K+1                                                        LOAD0070
        KK=J+3                                                       LOAD0080
        JJ=JJ+1                                                      LOAD0090
        AO(KK,I)=T(JJ)                                               LOAD0100
        AO(J,I)=T(K)                                                 LOAD0110
    1   CONTINUE                                                     LOAD0120
        RETURN                                                       LOAD0130
        END                                                          LOAD
```

Subroutine: LOADT

Purpose: To place unit initial conditions in the T block for the variational equations which are being used to generate the transition matrix.

Calling Sequence:

CALL LOADT

Input and Output

Common storages used or required: T

Subroutines required: None

Functions required: None

Approximate number of storages required:

LOADT

```
*       LABEL                                                            LOAT
*       SYMBOL TABLE                                                     LOAT
        SUBROUTINE LOADT                                                 LOAT0000
C       SUBROUTINE LOADT PLTS UNIT ICS ON PERTURBATION EQUATIONS         LOAT0010
        COMMON T                                                         LOAT0020
        DIMENSION T(1360),X(3,6),V(3,6)                                  LOAT0030
        EQUIVALENCE (T(7), X),(T(28),V)                                  LOAT0040
        DO 2 I=1,3                                                       LOAT0050
        L=I+3                                                            LOAT0060
        DO 1 J=1,3                                                       LOAT0070
        K=J+3                                                            LOAT0080
        X(I,J)=0.                                                        LOAT0090
        V(I,K)=0.                                                        LOAT0100
        X(I,K)=0.                                                        LOAT0110
        V(I,J)=0.                                                        LOAT0120
      1 CONTINUE                                                         LOAT0130
        X(I,I)=1.                                                        LOAT0140
        V(I,L)=1.                                                        LOAT0150
      2 CONTINUE                                                         LOAT0160
        RETURN                                                           LOAT0170
        END                                                              LOAT
```

Subroutine: MASS

Purpose: To find the relevant gravitational constants to be used in computing planetary perturbations for a given central body. MASS also chooses the initial integration step size as a function of central body.

Calling Sequence:

CALL MASS (NOR, UM, VKB, X, ACCP)

Input and **Output**

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	NOR	1			Central Body Indicator.
					1=Earth; 2, Moon; 3, Sun; 4,
					Venus; 5, Mars; 6, Jupiter
I	UM	6	μ_i	Km^3/sec^2	Gravitational constants, arranged
					as above
O	VKB	6	μ_i	Km^3/sec^2	μ_i for bodies
I	X	3	X,Y,Z	Km	Position coordinates
O	ACCP	1		seconds	Initial integration step size

Common storages used or required: None

Subroutines required: None

Functions required: FNORM

Approximate number of storages required: ________

Method of Establishing Integration Step Size and Perturbing Bodies

Central Body	Integration Step Size	Other Bodies to be Considered
Earth	60.0	Moon, Sun. If $\lvert X \rvert \geq 10^6$ Km, Jupiter.
Moon	60.0	Earth, Sun.
Sun	43200.0	All
Venus, Mars, Jupiter	60.0	All

The bodies which are not to be used in calculation of pertubation accelerations are eliminated from consideration by placing zeros in array VKB(6). For the bodies which are being considered, the appropriate gravitational constant is placed in VKB(6).

```
*      LABEL
*      SYMBOL TABLE
       SUBROUTINE MASS(NOR,UM,VKB,X,ACCP)
       DIMENSION UM(6),VKB(6),X(3)
       N = NOR
       DO 1 I=1,6
     1 VKB(I) = 0.
       GO TO (2,3,6,4,4,4),N
     2 VKB(2) = UM(2)
       ACCP = 60.
       DIS = FNORM(X)
       VKB(3) = UM(3)
       IF (DIS-1000000.) 11,10,10
    10 VKB(6) = UM(6)
       GO TO 12
    11 VKB(6) = 0.
    12 RETURN
     3 VKB(1) = UM(1)
       VKB(3) = UM(3)
       ACCP = 60.
       RETURN
     6 ACCP = 43200.
       GO TO 7
     4 ACCP = 60.
     7 DO 5 I=1,6
     5 VKB(I) = UM(I)
       VKB(N) = 0.
       RETURN
       END
```

Page Intentionally Left Blank

Subroutine: MATRX

Purpose: To perform matrix multiplications of matricies with any dimensions up to maximum dimensions of 10 by 10. A zero in the call sequence (J) yields an output matrix $C = A \cdot B$. A one in the call sequence (J) yields an output matrix $C = A \cdot B \cdot A^T$. The matrix products are obtained in double precision.

Calling Sequence:

CALL MATRX (A, B, C, NRA, NCA, NCB, J)

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	A	(10,10)			Input Matrix
I	B	(10,10)			Input Matrix
O	C	(10,10)			Output Matrix
I	NRA	1			Number of Rows of A
I	NCA	1			Number of Columns of A
I	NCB	1			Number of Columns of B
I	J	1			Type of multiplication desired

Common storages used or required: None

Subroutines required: None

Functions required: None

Approximate number of storages required:

```
*       LABEL                                                           MATX
*       SYMBOL TABLE                                                    MATX
CEC2000  SUBROUTINE MATRX                                               MATX
        SUBROUTINE MATRX(A,B,C,NRA,NCA,NCB,J)                           MATX0000
C       J=0 A*B=C                                                       MATX0010
C       J=1 A*B*AT=C  T MEANS TRANSPOSE                                 MATX0020
C       NRA=NUMBER OF ROWS OF A                                         MATX0030
C       NCA=COLUMNS OF A                                                MATX0040
C       NCB=COLUMNS OF B                                                MATX0050
C       ACCUMULATION OF PRODUCTS IN DOUBLE PRECISION                    MATX0060
        DIMENSION A(100),B(100),C(100),D(100)                           MATX0080
D       DUD=0.                                                          MATX0090
D       DUM=0.                                                          MATX0100
        DO 1 I=1,NRA                                                    MATX0110
        DO 1 K=1,NCB                                                    MATX0120
        NC=I+(K-1)*NRA                                                  MATX0130
D       CRUD=0.                                                         MATX0140
        DO 2 L=1,NCA                                                    MATX0150
        NA=I+(L-1)*NRA                                                  MATX0160
        NB=L+(K-1)*NCA                                                  MATX0170
        DUD=A(NA)                                                       MATX0180
        DUM=B(NB)                                                       MATX0190
D       CRUD=CRUD+DUD*DUM                                               MATX0200
      2 CONTINUE                                                        MATX0210
        D(NC)=CRUD                                                      MATX0220
      1 CONTINUE                                                        MATX0230
        IF(J)3,3,4                                                      MATX0240
      3 CONTINUE                                                        MATX0250
        KK=NCB*NRA                                                      MATX0260
        DO 5 I=1,KK                                                     MATX0270
      5 C(I)=D(I)                                                       MATX0280
        GO TO 10                                                        MATX0290
      4 CONTINUE                                                        MATX0300
        DO 6 I=1,NRA                                                    MATX0310
        DO 6 K=1,NRA                                                    MATX0320
        NC=I+(K-1)*NRA                                                  MATX0340
D       CRUD=0.                                                         MATX0350
        DO 7 L=1,NCA                                                    MATX0360
        NA=I+(L-1)*NRA                                                  MATX0370
        NB=K+(L-1)*NRA                                                  MATX0380
        DUM=D(NA)                                                       MATX0390
        DUD=A(NB)                                                       MATX0400
D       CRUD=CRUD+DUM*DUD                                               MATX0410
      7 CONTINUE                                                        MATX0420
        C(NC)=CRUD                                                      MATX0430
      6 CONTINUE                                                        MATX0440
     10 CONTINUE                                                        MATX0510
        RETURN                                                          MATX0520
        END                                                             MATX
```

Subroutine: MATSUB

Purpose: The subroutine is primarily logic which controls (1) trajectory and data printout, (2) updating of the state covariance matrix for observations by calling subroutines EARTR, ONBTR, and MONBTR, and (3) updating of the guidance covariance matrix by calling subroutine GUID.

Calling Sequence:

CALL MATSUB

Input and Output

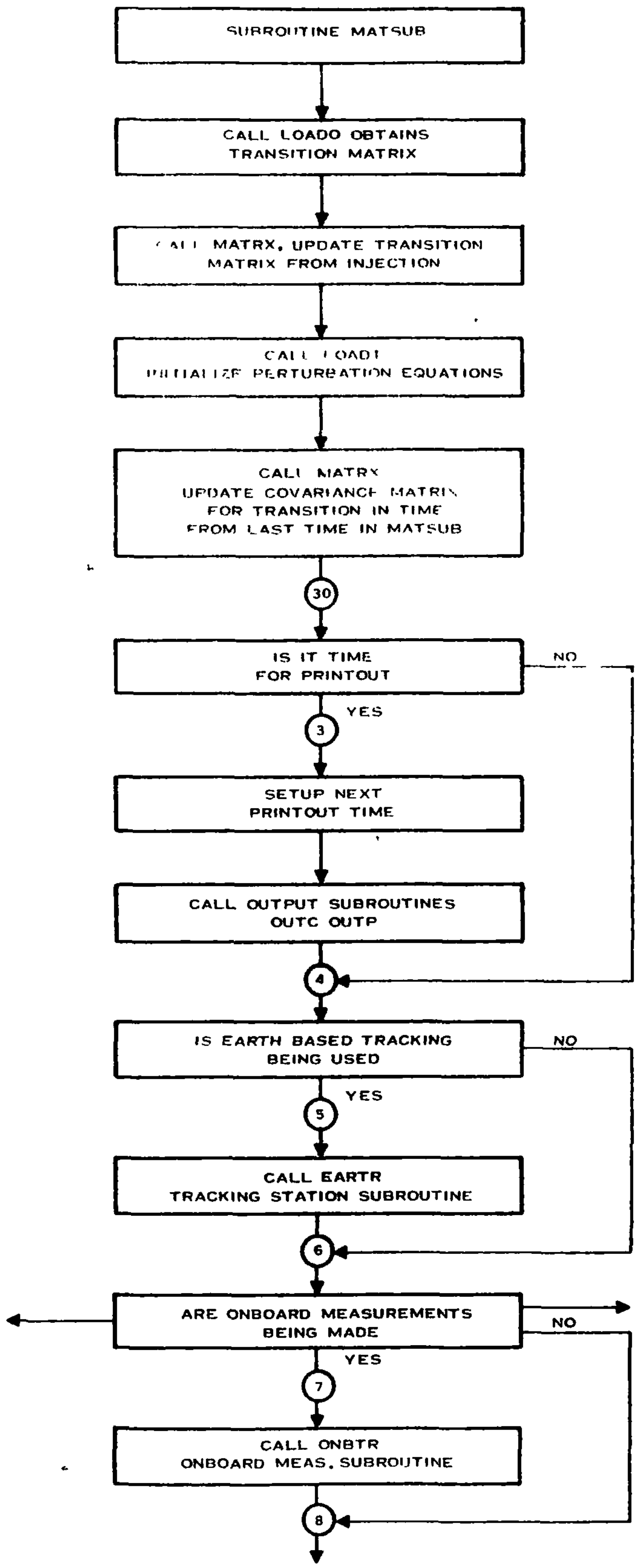
I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I					Quantities obtained from common
O					Quantities returned to common

Common storages used or required: T, S, C, IC

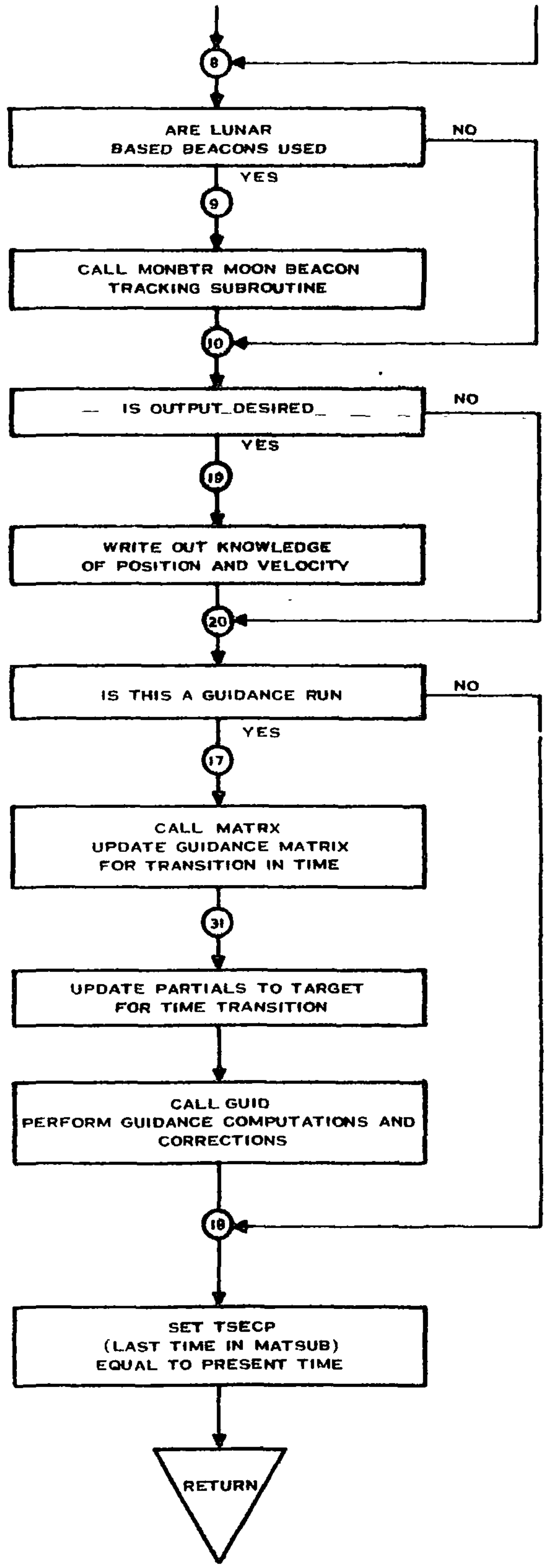
Subroutines required: EARTR, GUID, INVAO, LOADO, LOADT, MATRX, MONBTR, ONBTR, OUTC, OUTP

Functions required: SQRT (FIL)(STH)

Approximate number of storages required: 346 DEC


SUBROUTINE MATSUB
CALL LOADO OBTAINS TRANSITION MATRIX
CALL MATRX, UPDATE TRANSITION MATRIX FROM INJECTION
CALL LOADI INITIALIZE PERTURBATION EQUATIONS
CALL MATRX UPDATE COVARIANCE MATRIX FOR TRANSITION IN TIME FROM LAST TIME IN MATSUB
30
IS IT TIME FOR PRINTOUT
NO
YES
3
SETUP NEXT PRINTOUT TIME
CALL OUTPUT SUBROUTINES OUTC OUTP
4
IS EARTH BASED TRACKING BEING USED
NO
YES
5
CALL EARTR TRACKING STATION SUBROUTINE
6
ARE ONBOARD MEASUREMENTS BEING MADE
NO
YES
7
CALL ONBTR ONBOARD MEAS. SUBROUTINE
8

MATSUB -2


8
ARE LUNAR BASED BEACONS USED
NO
YES
9
CALL MONBTR MOON BEACON TRACKING SUBROUTINE
10
IS OUTPUT DESIRED
NO
YES
19
WRITE OUT KNOWLEDGE OF POSITION AND VELOCITY
20
IS THIS A GUIDANCE RUN
NO
YES
17
CALL MATRX UPDATE GUIDANCE MATRIX FOR TRANSITION IN TIME
31
UPDATE PARTIALS TO TARGET FOR TIME TRANSITION
CALL GUID PERFORM GUIDANCE COMPUTATIONS AND CORRECTIONS
18
SET TSECP (LAST TIME IN MATSUB) EQUAL TO PRESENT TIME
RETURN

```
*     LABEL                                                                MATS
*     SYMBOL TABLE                                                         MATS
CEC2006  SUBROUTINE MATSUB                                                 MATS
      SUBROUTINE MATSUB                                                    MATS0000
      COMMON T,S,C,IC                                                      MATS0010
      DIMENSION T(1360),S(1000),C(1000),IC(1),AN(3,3),PO(22),VE(22),       MATS0020
     1EM(3,3),XP(3),VXP(3),EA(3,3),XED(3),VED(3),AO(6,6),P(6,6),           MATS0030
     2TFTA(3,6),PFTA(6,6),AOS(6,6),AOI(6,6)                                MATS0040
      EQUIVALENCE (T ,TDUM),(S ,SDUM),(C ,CDUM),                           MATS0050
     1(C(138),AN),(C(13),TP1),(C(14),TP2),(C(62),PO),(C(84),VE),           MATS0060
     2(C(120),EM),(C(650),TPRINT),(C(651),TSTRT),(C(30),TSEC),             MATS0070
     3(C(15),XP),(C(18),VXP),(C(10),TW),(C(11),TF) ,(C(129),EA),           MATS0080
     4(C(652),P),(C(752),AOS),(C(850),AOI)                                 MATS0090
     5,(C(460),PFTA),(C(568),TFTA),(C(647),TOUT)                           MATS0100
      EQUIVALENCE (IC(2),IOR),(IC(7),KOUT),(IC(6),ITARG),(IC(9),LPRINT),MATS0110
     1(IC(190),LSTAT),(IC(191),LONB),(IC(192),LMB),(IC(193),IGUID),        MATS0120
     2(IC(214),NOUT),(C(649),TSECP)                                        MATS0130
      CALL SETN(NIN,NUTS)                                                  MATS0131
      NOUT=1                                                               MATS0140
      CALL LOADO(AO)                                                       MATS0145
      CALL MATRX(AO,AOS,AOS,6,6,6,0)                                       MATS0150
      CALL LOADT                                                           MATS0155
      CALL MATRX(AO,P,P,6,6,6,1)                                           MATS0160
      DO 30 I=1,6                                                          MATS0165
      DO 30 J=I,6                                                          MATS0170
      P(I,J)=(P(I,J)+P(J,I))/2.                                            MATS0175
      P(J,I)=P(I,J)                                                        MATS0180
   30 CONTINUE                                                             MATS0185
      IF(TOUT-TSEC) 3,3,4                                                  MATS0220
    3 CONTINUE                                                             MATS0230
      TOUT=TOUT+TPRINT                                                     MATS0240
      NOUT=2                                                               MATS0250
C     NOUT IS USED TO INDICATE OTHER OUTPUT WANTED WHEN = 2                MATS0260
      CALL OUTC                                                            MATS0270
      CALL OUTP                                                            MATS0280
    4 CONTINUE                                                             MATS0290
C     LSTAT IS SET 2 BY INPUT CARDS IF ANY EARTH BASED TRACKING            MATS0300
      LSTAT=LSTAT                                                          MATS0310
      GO TO (6,5),LSTAT                                                    MATS0320
    5 CONTINUE                                                             MATS0330
      CALL EARTR                                                           MATS0340
C     EARTR UDDATS COV MATRIX P FOR EARTH BASED TRACKING                   MATS0350
    6 CONTINUE                                                             MATS0360
      LONB=LONB                                                            MATS0370
      GO TO (8,7),LONB                                                     MATS0380
C     LONB IS SET 2 BY INPUT CARDS IF ANY ONBOARD MEASUREMENTS MADE        MATS0390
    7 CONTINUE                                                             MATS0400
C     ONBTR  UPDATES COV MATRIX P FOR THE ONBOARD MEASUREMENTS             MATS0410
      CALL ONBTR                                                           MATS0420
    8 CONTINUE                                                             MATS0430
      LMB=LMB                                                              MATS0440
```

MATSUB (Cont'd)

```
      GO TO (10,9),LMB                                                  MATS0450
C     LMB IS SET 2 BY INPUT CARDS IF MOON BASED BEACONS USED            MATS0460
    9 CONTINUE                                                          MATS0470
      CALL MONBTR                                                       MATS0480
C    MONBTR UPDATS COV MATRIX P FOR BEACONS ON MOON                     MATS0490
   10 CONTINUE                                                          MATS0500
      GO TO (20,19), NOUT                                               MATS0510
  19  CONTINUE                                                          MATS0520
      RMSP=SQRTF (P (1,1)+P (2,2)+P (3,3))                              MATS0530
      RMSVP=SQRTF(P (4,4)+P (5,5)+P (6,6))                              MATS0540
      WRITE OUTPUT TAPE NUTS,700,RMSP,RMSVP                             MATS0550
 700  FORMAT(42H KNOWLEDGE OF STATE AFTER ALL OBSERVATIONS,/            MATS0560
     124H            RMS POSITION=,E15.8,25H            RMS VELOCITY=,  MATS0570
     2E15.8)                                                            MATS0580
  20  CONTINUE                                                          MATS0590
C     NUMBERS 11-16 RESERVED FOR FUTURE MEASUREMENTS                    MATS0600
      IGUID=IGUID                                                       MATS0610
      GO TO (18,17),IGUID                                               MATS0620
C     IGUID IS SET BY INPUT TO MAIN PROGRAM IF GUIDANCE CALCULATIONS AREMATS0630
C     MADE                                                              MATS0640
   17 CONTINUE                                                          MATS0650
      CALL MATRX(AO,PFTA,PFTA,6,6,6,1)                                  MATS0655
      DO 31 I=1,6                                                       MATS0660
      DO 31 J=I,6                                                       MATS0665
      PFTA(I,J)=(PFTA(I,J)+PFTA(J,I))/2.                                MATS0670
      PFTA(J,I)=PFTA(I,J)                                               MATS0675
  31  CONTINUE                                                          MATS0680
      CALL INVAO(AO,AOI)                                                MATS0685
      CALL MATRX(TFTA,AOI,TFTA,3,6,6,0)                                 MATS0690
      CALL GUID                                                         MATS0710
C     GUID PERFORMS THE GUIDANCE CALCULATIONS                           MATS0720
   18 CONTINUE                                                          MATS0730
      TSECP=TSEC                                                        MATS0740
      RETURN                                                            MATS0750
      END                                                               MATS
```

Page Intentionally Left Blank

Subroutine: MNA

Purpose: To provide the rotation matrix EMN which transforms moon centered coordinates in the earth's true equator and equinox to moon centered coordinates in the moon's true equator.

Calling Sequence:

CALL MNA (TIME, OM, CR, DT, EPSIL, RO, G, GP, WW, EMN)

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	TIME	1	d	days	Days since 1950.0
I	OM	1	Ω		Output from NUTAIT: (Arg. De-
					cending node)
I	CR	1	☾	radians	Output from NATAIT: (mean long.
					of the moon)
I	DT	1	$\Delta\Psi + d\Psi$	radians	Output from NUTAIT: (nutation in
					longitude)
I	EPSIL	1	$\bar{\epsilon} + \Delta\epsilon + d\epsilon$	radians	Output from NUTAIT: (mean obliquity
					and nutation in obliquity)

(Cont'd)

Common storages used or required: None.

Subroutines required: System double precision routines.

Functions required: INTF, SQRT, SIN, COS, ACOS

Approximate number of storages required:

(Cont'd.)

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
O	RO	1	ρ		Libration term
O	G	1	g	radians	Mean anomaly of the moon
O	GP	1	g	radians	Mean anomaly of the sun
O	WW	1	ω	radians	Argument of the perigee of the
					moon
O	EMN	3,3			Rotation matrix

Transformation From Earth's True Equator to Moon's True Equator.

The two rectangular systems are related through Λ, Ω', and i by the rotation:

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix}_{\text{MOON}} = \begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{pmatrix} \begin{pmatrix} X' \\ Y' \\ Z' \end{pmatrix}_{\text{EARTH}}$$

where

$$b_{11} = \cos \Lambda \cos \Omega' - \sin \Lambda \sin \Omega' \cos i$$

$$b_{12} = \cos \Lambda \sin \Omega' + \sin \Lambda \cos \Omega' \cos i$$

$$b_{13} = \cos \Lambda \sin i$$

$$b_{21} = -\sin \Lambda \cos \Omega' - \cos \Lambda \sin \Omega' \cos i$$

$$b_{22} = -\sin\Lambda \sin\Omega' + \cos\Lambda \cos\Omega' \cos i$$

$$b_{23} = \cos\Lambda \sin i$$

$$b_{31} = \sin\Omega' \sin i$$

$$b_{32} = -\cos\Omega' \sin i$$

$$b_{33} = \cos i$$

i is the inclination of the moon's true equator to the earth's equator

Ω' is the right ascension of the ascending node of the moon's true equator

Λ is the anomaly from the node to the X axis

$$\Lambda = \Delta + (☾ + \Upsilon) - (\Omega + \sigma)$$

Δ is the anomaly from the node to the ascending node of the moon's true equator on the ecliptic

Ω is the mean longitude of the descending node of the moon's mean equator on the ecliptic

☾ is the mean longitude of the moon

σ is the libration in the node

Υ is the libration in the mean longitude

ρ is the libration in the inclination.

$\delta\Psi$, ϵ, Ω, and ☾ are input quantities obtained from NUTAIT. The remainder are computed from the following equations.

I = inclination of moon's equator to ecliptic

$I = 1.535^{\circ}$

g = mean anomaly of moon

$g = 215.54013 + 13.064992\ d$

g' = mean anomaly of sun

g' = $358.009067 + .9856005\ d$

ω = argument of perigee of moon

ω = $196.745632 + .1643586\ d$

where d = days from 1950.

$$\sigma \sin I = -.0302777 \sin g + .0102777 \sin (g + 2\omega) - .00305555 \sin (2g + 2\omega)$$

$$\tau = -.003333 \sin g + .0163888 \sin g' + .005 \sin 2\omega$$

$$\rho = -.0297222 \cos g + .0102777 \cos (g + 2\omega) - .00305555 \cos (2g + 2\omega)$$

$$\cos i = \cos (\Omega + \sigma + \delta\Psi) \sin \epsilon \sin (I + \rho) + \cos \epsilon \cos (I + \rho) \qquad 0 < i < 90^\circ$$

$$\sin \Omega' = -\sin (\Omega + \sigma + \delta\Psi) \sin (I + \rho) \csc i \qquad -90^\circ < \Omega' < 90^\circ$$

$$\sin \Delta = -\sin (\Omega + \sigma + \delta\Psi) \sin \epsilon \csc i$$

$$\cos \Delta = -\sin (\Omega + \sigma + \delta\Psi) \sin \Omega' \cos \epsilon - \cos (\Omega + \sigma + \delta\Psi) \cos \Omega' \qquad 0^\circ \le \Delta < 360^\circ$$

Reference: JPL Technical Report No. 32-223

MNA

```
*       LABEL
*       SYMBOL TABLE
 CEC20AH
        SUBROUTINE MNA(TIME,OM,CR,DT,EPSIL,RO,G,GP,WW,EM)
        DIMENSION EM(3,3),DF(3)
        D = TIME
        T = D/36525.
        T2 = T*T
        T3 = T2*T
        A=13.064992
        DO 6 I=1,3
        DD=D
 D      DD=DD*(A/360.)
 D      DD=DD-INTF(DD)
        DF(I)=DD
        GO TO (4,5,6),I
    4   A=.9856005
        GO TO 6
    5   A=.1643586
    6   CONTINUE
        G=215.54013+360.*DF(1)
        GP=358.009067+360.*DF(2)
        WW=196.745632+360.*DF(3)
        G = G*.017453296
        GP = GP*.017453296
        WW = WW*.017453296
        YN = 1.535*.017453296
        RO = -.0297222*COSF(G) + .01020777*COSF(G+2.*WW)
       1 -.00305555*COSF(2.*G+2.*WW)
        TA = -.003333*SINF(G) + .0163888*SINF(GP)
       1 +.005*SINF(2.*WW)
        SG = -.0302777*SINF(G) + .0102777*SINF(G+2.*WW)
       1 -.00305555*SINF(2.*G+2.*WW)
        SG = (SG*.017453296)/SINF(YN)
        RO = RO*.017453296
        TA = TA*.017453296
        YN = YN + RO
        RO = OM + SG + DT
        CI = COSF(RO)*SINF(EPSIL)*SINF(YN)
       1 +COSF(EPSIL)*COSF(YN)
        SI = 1. - CI**2
        SI = SQRTF(SI)
        SO = -SINF(RO)*SINF(YN)/SI
        CO = 1. - SO**2
        CO = SQRTF(CO)
        SD = -SINF(RO)*SINF(EPSIL)/SI
        CD = -SINF(RO)*SO*COSF(EPSIL) - COSF(RO)*CO
        DL = ACOSF(CD)
        IF(SD)1,3,3
  1     DL = 6.283185306 - DL
  3     CONTINUE
```


PHILCO
A SUBSIDIARY OF Ford Motor Company,
WDL DIVISION

MNA (Cont'd)

```
CA = DL + (CR + TA) - (OM + SG)                                  SMNA048
SA = SINF(CA)                                                    SMNA049
CA = COSF(CA)                                                    SMNA050
RO = COSF(RO)*SINF(EPSIL)/(SI*CD)                                SMNA051
EM(1,1) = CA*CO - SA*SO*CI                                       SMNA052
EM(1,2) = CA*SO + SA*CO*CI                                       SMNA053
EM(1,3) = SA*SI                                                  SMNA054
EM(2,1) = -SA*CO - CA*SO*CI                                      SMNA055
EM(2,2) = -SA*SO + CA*CO*CI                                      SMNA056
EM(2,3) = CA*SI                                                  SMNA057
EM(3,1) = SO*SI                                                  SMNA058
EM(3,2) = -CO*SI                                                 SMNA059
EM(3,3) = CI                                                     SMNA060
RETURN                                                           SMNA061
END                                                              SMNA
```

Subroutine: MNAND

Purpose: To compute the matrix M used to perform the velocity transformation corresponding to the position transformation described in MNA.

Calling Sequence:

CALL MNAND (TIME, RO, G, GP, WW, EM, DM)

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	TIME	1	d	days	Days since 1950.0
I	RO	1	ρ	radians	Output from MNA; see description
I	G	1	g	radians	Output from MNA; see description
I	GP	1	g'	radians	Output from MNA; see description
I	WW	1	ω	radians	Output from MNA; see description
I	EM	3,3	M		Output from MNA; matrix M
O	DM	3,3	$\dot{M}$		Matrix $\dot{M}$

Common storages used or required: None

Subroutines required: None

Functions required: COS

Approximate number of storages required:

Theory and Equations

In transforming lunacentric position coordinates relative to the earth's equator and equinox of 1950.0 to coordinates relative to the moon's true equator, three matrices are computed. Matrix A (from ROTEQ) rotates the coordinates to equinox of date, matrix N (from NUTAIT) accounts for the nutation of the earth about its precessing mean equator, and matrix M (from MNA) transforms to the position relative to the moon's true equator.

Then
$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix}_{\text{moon}} = MNA \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}_{\text{earth, 1950.0}}$$

Assuming that $\dot{N} = \dot{A} = 0$

$$\begin{pmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{pmatrix}_{\text{moon}} = MNA \begin{pmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{pmatrix}_{\text{earth, 1950}} + \dot{M}NA \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}_{\text{earth, 1950}}$$

The $\dot{M}$ matrix is computed from the equations:

$$\dot{M} = \begin{pmatrix} M_{21}\dot{\Lambda} & M_{22}\dot{\Lambda} & M_{23}\dot{\Lambda} \\ -M_{11}\dot{\Lambda} & -M_{12}\dot{\Lambda} & -M_{13}\dot{\Lambda} \\ 0 & 0 & 0 \end{pmatrix}$$

where M_{ij} are the elements of M, and

$$\dot{\Lambda} = \dot{\Delta} + \dot{☾} + \dot{t} - \dot{\Omega} - \dot{\sigma}$$

$$\dot{\Delta} = -\rho\ (\dot{\Omega} + \dot{\sigma})$$

$$\dot{☾} = 0.266170762 \times 10^{-5} - 0.12499171 \times 10^{-13}\ T \text{ rad/sec};\ T = d/36525$$

$$\dot{\Omega} = -0.1069698435 \times 10^{-7} + 0.23015329 \times 10^{-3}\ T \text{ rad/sec}$$

$$\dot{t} = -0.1535272946 \times 10^{-9} \cos g$$
$$+0.569494067 \times 10^{-10} \cos g$$
$$+0.579473484 \times 10^{-11} \cos 2\omega \text{ rad/sec}$$

$$\dot{\sigma} = -0.520642191 \times 10^{-7} \cos g$$
$$+0.1811774451 \times 10^{-7} \cos (g+2\omega)$$
$$-0.1064057858 \times 10^{-7} \cos (2\omega+2g) \text{ rad/sec}$$

Reference: JPL Technical Report No. 32-223

MNAND

```
*       LABEL
*       SYMBOL TABLE
        SUBROUTINE MNAND(TIME,RO,G,GP,WW,EM,DM)
        DIMENSION DM(3,3),EM(3,3)
        T = TIME/36525.
        CRD = .266170762E-5 - .12499171E-13*T
        OMD = -.1069698435E-7 + .23015329E-13*T
        TAD = -1.535272946*COSF(G) + .569494067*COSF(GP)
       1 + .0579473484*COSF(2.*WW)
        TAD = .1E-9*TAD
        SGD = -.520642191*COSF(G) + .1811774451*COSF(G+2.*WW)
       1 -.1064057858*COSF(2.*WW+2.*G)
        SGD = .1E-6*SGD
        DLD = -RO*(OMD + SGD)
        CAD = DLD + CRD + TAD - OMD - SGD
        DM(1,1) = EM(2,1)*CAD
        DM(1,2) = EM(2,2)*CAD
        DM(1,3) = EM(2,3)*CAD
        DM(2,1) = -EM(1,1)*CAD
        DM(2,2) = -EM(1,2)*CAD
        DM(2,3) = -EM(1,3)*CAD
        DM(3,1) = 0.0
        DM(3,2) = 0.0
        DM(3,3) = 0.0
        RETURN
        END
```

Subroutine: MONBTR

Purpose: To obtain the position and velocity vectors of the vehicle relative to the beacon and determine if the beacon is in view of the vehicle. The station location partials are computed and subroutine COMPHQ called to update the state covariance matrix for the measurements being made. The types of measurements possible are range, range rate, right ascension, and declination.

Calling Sequence:

CALL MONBTR

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I					Quantities obtained from common
O					Quantities placed in common

Common storages used or required: T, S, C, IC

Subroutines required: COMPHO, DOT, INTRI, MATRX, MNAND, MNA, NUTAIT, TRAC

Functions required: ASIN, ATAN, FNORM, SQRT (FIL)(SLO)(STH)

Approximate number of storages required: 495 DEC

MONBTR Partial Derivatives

MONBTR computes two partial matrices which are used in COMPHQ to evaluate station location errors. The matrices are the following:

$$\text{STPARS} = \begin{pmatrix} \frac{\partial x_1}{\partial LAT} & \frac{\partial x_1}{\partial LON} & \frac{\partial x_1}{\partial ALT} \\ \frac{\partial x_1}{\partial LAT} & \frac{\partial x_2}{\partial LON} & \frac{\partial x_2}{\partial ALT} \\ \frac{\partial x_3}{\partial LAT} & \frac{\partial x_3}{\partial LON} & \frac{\partial x_3}{\partial ALT} \end{pmatrix}$$

$$\text{STPARD} = \begin{pmatrix} \frac{\partial \dot{x}_1}{\partial LAT} & \frac{\partial \dot{x}_1}{\partial LON} & \frac{\partial \dot{x}_1}{\partial ALT} \\ \frac{\partial \dot{x}_2}{\partial LAT} & \frac{\partial \dot{x}_2}{\partial LON} & \frac{\partial \dot{x}_2}{\partial ALT} \\ \frac{\partial \dot{x}_3}{\partial LAT} & \frac{\partial \dot{x}_3}{\partial LON} & \frac{\partial \dot{x}_3}{\partial ALT} \end{pmatrix}$$

where $\vec{X} = \begin{pmatrix} X_1 \\ X_2 \\ X_3 \end{pmatrix} = -\vec{X}_V + \vec{X}_B \equiv$ XREL (FORTRAN)

$$\dot{\vec{X}} = \begin{pmatrix} \dot{X}_1 \\ \dot{X}_2 \\ \dot{X}_3 \end{pmatrix} = -\dot{\vec{X}}_V + \dot{\vec{X}}_B \equiv \text{VREL (FORTRAN)}$$

$$\frac{\partial x_i}{\partial LAT} = \frac{\partial x_{Bi}}{\partial LAT} \qquad \frac{\partial \dot{x}_i}{\partial LAT} = \frac{\partial \dot{x}_{Bi}}{\partial LAT} \qquad i = 1,2,3$$

$$\vec{X}_B = (MNA)^T \; \vec{X}_{BM}$$

$$\dot{\vec{X}}_B = (MNAND)^T \; \vec{X}_{BM} + \overset{(=0)}{\cancel{(MNA)^T \dot{\vec{X}}_{BM}}}$$

where subscripts BM indicate beacon moon-fixed coordinates.

The transformations MNA and MNAND are obtained by calling the subroutines MNA and MNAND.

$$\underset{3 \times 1}{\vec{X}_{BM}} = \underset{3 \times 3}{(R + ALT)} \underset{3 \times 1}{\begin{pmatrix} \cos LAT \cos LON \\ \cos LAT \sin LON \\ \sin LAT \end{pmatrix}}$$

$$\begin{pmatrix} \frac{\partial x_1}{\partial PAR} \\ \frac{\partial x_2}{\partial PAR} \\ \frac{\partial x_3}{\partial PAR} \end{pmatrix} = (MNA)^T \begin{pmatrix} \frac{\partial X_{BM}}{\partial PAR} \\ \frac{\partial Y_{BM}}{\partial PAR} \\ \frac{\partial Z_{BM}}{\partial PAR} \end{pmatrix}$$

$$\begin{pmatrix} \frac{\partial \dot{x}_1}{\partial PAR} \\ \frac{\partial \dot{x}_2}{\partial PAR} \\ \frac{\partial \dot{x}_3}{\partial PAR} \end{pmatrix} = (MNAND)^T \begin{pmatrix} \frac{\partial X_{BM}}{\partial PAR} \\ \frac{\partial Y_{BM}}{\partial PAR} \\ \frac{\partial Z_{BM}}{\partial PAR} \end{pmatrix}$$

$$\underset{3 \times 1}{\frac{\partial \vec{X}_{BM}}{\partial LAT}} = (R + ALT) \underset{3 \times 1}{\begin{pmatrix} -\sin LAT \cos LON \\ -\sin LAT \sin LON \\ \cos LAT \end{pmatrix}}$$

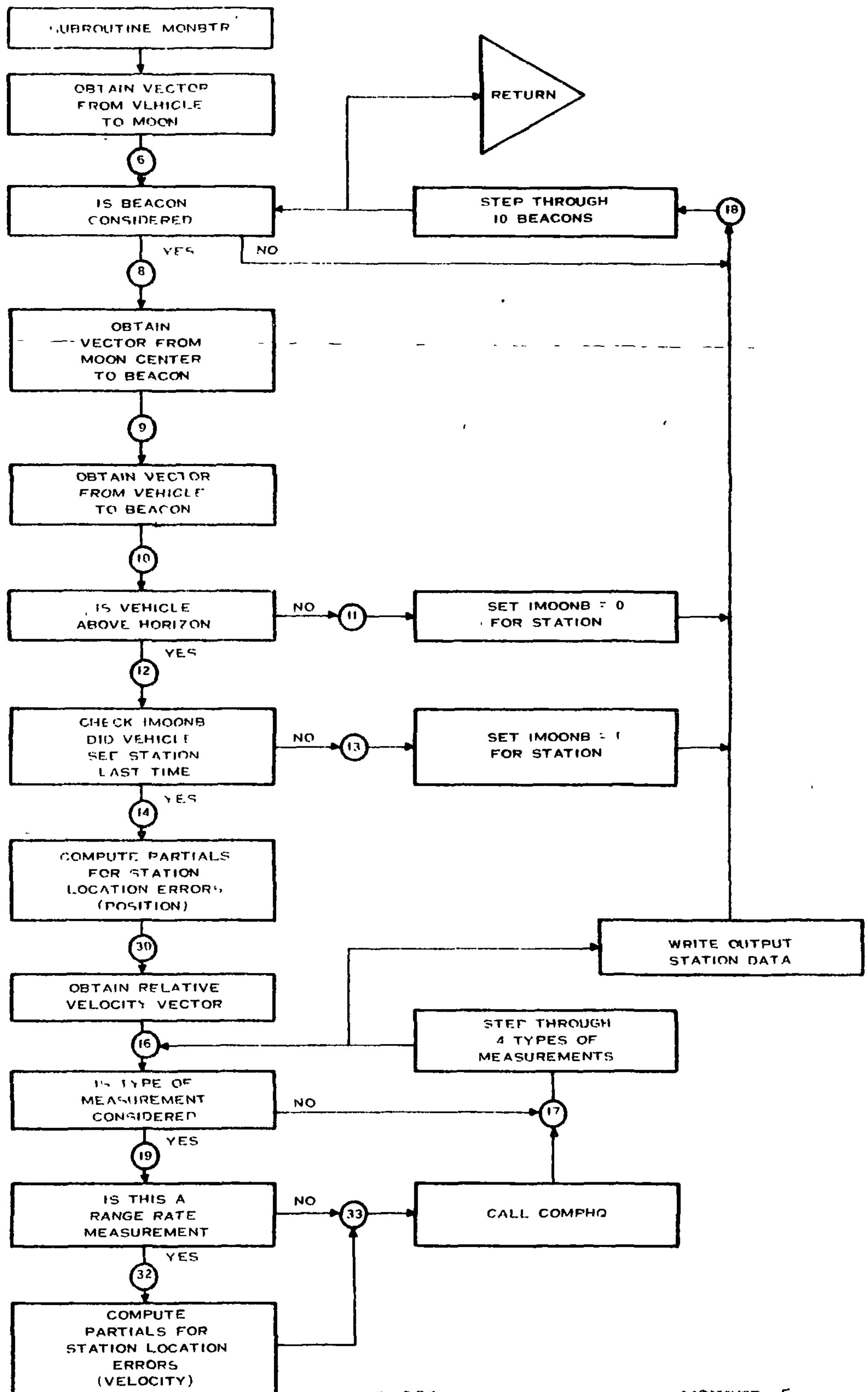
$$\underset{3 \times 1}{\frac{\partial \vec{X}_{BM}}{\partial LON}} = (R + ALT) \underset{3 \times 1}{\begin{pmatrix} -\cos LAT \sin LON \\ \cos LAT \cos LON \\ 0 \end{pmatrix}}$$

$$\underset{3 \times 1}{\frac{\partial \vec{X}_{BM}}{\partial ALT}} = \underset{3 \times 1}{\begin{pmatrix} \cos LAT \cos LON \\ \cos LAT \sin LON \\ \sin LAT \end{pmatrix}}$$

$$\underset{3 \times 3}{STPARS} = \underset{3 \times 3}{(MNA)^T} \underset{3 \times 3}{\left(\frac{\partial \vec{X}_{BM}}{\partial LAT} ; \frac{\partial \vec{X}_{BM}}{\partial LON} ; \frac{\partial \vec{X}_{BM}}{\partial ALT} \right)}$$

$$\underset{3 \times 3}{STPARD} = \underset{3 \times 3}{(MNAND)^T} \underset{3 \times 3}{\left(\frac{\partial \vec{X}_{BM}}{\partial LAT} ; \frac{\partial \vec{X}_{BM}}{\partial LON} ; \frac{\partial \vec{X}_{BM}}{\partial ALT} \right)}$$

Following computation of the matricies, subroutine COMPHQ is called to perform the updating of the state covariance matrix.


SUBROUTINE MONBTR
OBTAIN VECTOR FROM VEHICLE TO MOON
RETURN
IS BEACON CONSIDERED
STEP THROUGH 10 BEACONS
YES
NO
OBTAIN VECTOR FROM MOON CENTER TO BEACON
OBTAIN VECTOR FROM VEHICLE TO BEACON
IS VEHICLE ABOVE HORIZON
SET IMOONB = 0 FOR STATION
CHECK IMOONB DID VEHICLE SEE STATION LAST TIME
SET IMOONB = 1 FOR STATION
COMPUTE PARTIALS FOR STATION LOCATION ERRORS (POSITION)
WRITE OUTPUT STATION DATA
OBTAIN RELATIVE VELOCITY VECTOR
STEP THROUGH 4 TYPES OF MEASUREMENTS
IS TYPE OF MEASUREMENT CONSIDERED
IS THIS A RANGE RATE MEASUREMENT
CALL COMPHO
COMPUTE PARTIALS FOR STATION LOCATION ERRORS (VELOCITY)

MONBTR

```
*       LABEL
*       SYMBOL TABLE
CEC2012  SUBROUTINE MONBTR
        SUBROUTINE MONBTR
        COMMON T,S,C,IC
        DIMENSION
       1T(1360),S(1000),C(1000),IC(1),XREL(3),VREL(3)
       1,PO(22),VE(22),AN(3,3),EM(3,3),ISBCON(10),U(3),EN(3)
       2,RT(3),E(3),ETM(3,3),IMOONB(10),DM(3,3),DUM(3,3)
       3,RMB(3),TYMEAS(4,10),X(3),VX(3),OUTPUT(6)
        DIMENSION STPARS(3,3),DUD(3,3),STPARD(3,3)
        EQUIVALENCE (T,TDUM),(S,SDUM),(C,CDUM),(IC,ICDUM)
       1,(C(895),XREL),(C(898),VREL),(C(894), OBNO),(C(893),XMAG),
       2(C(892),DEX),(C(891),DEN1),(C(890),DEN2)
       3,(C(889),DEX2),(C(888),DEX3),(C(652),P),(C(973),OUTPUT)
       1,(C(13),TW),(C(14),TF),(C(62),PO),(IC(3),NOR)
       2,(IC(190),LSTAT),(C(138),AN),(S(24),A),(C(120),EM)
       3,(IC(140),ISBCON),(S(72),DR),(IC(220),IMOONB),(C(649),TSECP)
       4,(C(30),TSEC),(IC(140),TYMEAS),(C(15),X),(C(18),VX)
        EQUIVALENCE (C(800),STPARD),(C(788),STPARS)
        CALL SETN(NIN,NUTS)
        DELTT=TSEC-TSECP
        LSTAT=LSTAT
        GO TO(1,2),LSTAT
  1     CONTINUE
        NB=NOR-1
        DIS=1.E10
        CALL INTR1(TW,TF,NB,PO,1,VE,DIS)
  2     CONTINUE
        TIME=TW+TF
        CALL NUTAIT(TIME,OM,CR,DT,EM,EPSIL)
        CALL MNA(TIME,OM,CR,DT,EPSIL,RO,G,GP,WW,ETM)
        CALL MNAND(TIME,RO,G,GP,WW,ETM,DM)
        IF(NOR-2)3,5,3
  3     CONTINUE
        DO 4 I=1,3
        J=20-I
        XREL(I)=PO(J)-X(I)
        VREL(I)=VE(J)-VX(I)
  4     CONTINUE
        GO TO 7
  5     CONTINUE
        DO 6 I=1,3
        XREL(I)=-X(I)
        VREL(I)=-VX(I)
  6     CONTINUE
C       XREL POSITION VECTOR FROM VEHICLE TO MOON CENTER 1950
C       VREL RELATIVE VELOCITY OF MOON CENTER 1950
  7     CONTINUE
        NN=485
        DO 18 III=1,10
```

```
      NN=NN+15                                                          MONB045
      IF(ISBCON(III))18,18,8                                            MONB046
    8 CONTINUE                                                          MONB047
      AT=S(NN+9)*DR                                                     MONB048
      ON=S(NN+10)*DR                                                    MONB049
      AL=S(NN+11)                                                       MONB050
      SNAME=S(NN+12)                                                    MONB051
      GH=0.                                                             MONB052
      CALL TRAC(ON,AT,AL,GH,U,E,EN,RT,AC,SL,CL,ST,CT,A,A)               MONB053
      CALL MATRX(ETM,AN,DUM,3,3,3,0)                                    MONB054
      DO 9 I=1,3                                                        MONB055
      RMB(I)=0.                                                         MONB056
      DO 9 J=1,3                                                        MONB057
      RMB(I)=RMB(I)+DUM(J,I)*RT(J)                                      MONB058
    9 CONTINUE                                                          MONB059
C     RMB RADIUS VECTOR TO MOON BEACON 1950                             MONB060
      DO 10 I=1,3                                                       MONB061
      XREL(I)=XREL(I)+RMB(I)                                            MONB062
C     XREL VECTOR FROM VEHICLE TO MOON BEACON 1950                      MONB063
   10 CONTINUE                                                          MONB064
      TEST=DOT(RMB,XREL)                                                MONB065
      IF(TEST)12,12,11                                                  MONB066
   11 IMOONB(III)=0                                                     MONB067
C     VEHICLE BELOW BEACON HORIZON                                      MONB068
      GO TO 18                                                          MONB069
   12 CONTINUE                                                          MONB070
      IF(IMOONB(III))13,13,14                                           MONB071
   13 CONTINUE                                                          MONB072
      IMOONB(III)=1                                                     MONB073
      GO TO 18                                                          MONB074
   14 CONTINUE                                                          MONB074
      RADIUS=A+AL                                                       MONB074
      DUD(1,1)=-RADIUS*SL*CT                                            MONB074
      DUD(2,1)=-RADIUS*SL*ST                                            MONB074
      DUD(3,1)=RADIUS*CL                                                MONB074
      DUD(1,2)=-RADIUS*CL*ST                                            MONB074
      DUD(2,2)=RADIUS*CL*CT                                             MONB074
      DUD(3,2)=0.                                                       MONB074
      DUD(1,3)=CL*CT                                                    MONB074
      DUD(2,3)=CL*ST                                                    MONB075
      DUD(3,3)=SL                                                       MONB075
      DO 30 I=1,3                                                       MONB075
      DO 30 J=1,3                                                       MONB075
      STPARS(I,J)=0.                                                    MONB075
      DO 30 K=1,3                                                       MONB075
   30 STPARS(I,J)=STPARS(I,J)+DUM(K,I)*DUD(K,J)                         MONB075
      CALL MATRX(DM,AN,DUM,3,3,3,0)                                     MONB076
      DO 15 I=1,3                                                       MONB077
      RMB(I)=0                                                          MONB078
      DO 15 J=1,3                                                       MONB079
      RMB(I)=RMB(I)+DUM(J,I)*RT(J)                                      MONB080
```

MONBTR (Cont'd)

```
 15   CONTINUE                                                          MONB0810
C     RMB ROTATIONAL VELOCITY OF MOON BEACON 1950                       MONB0820
      DO 16 I=1,3                                                       MONB0830
      VREL(I)=VREL(I)+RMB(I)                                            MONB0840
C     VREL RELATIVE VELOCITY OF MOON BEACON WRT THE VEHICLE             MONB0850
 16   CONTINUE                                                          MONB0860
      OBNO=DELTT/S(NN+13)                                               MONB0870
      XMAG=FNORM(XREL)                                                  MONB0880
      DEX=XREL(1)**2+XREL(2)**2                                         MONB0890
      DEN1=1./SQRTF(DEX)/XMAG**2                                        MONB0900
      DEN2=1./DEX                                                       MONB0910
      DEX2=XMAG*XMAG                                                    MONB0920
      DEX3=1./XMAG                                                      MONB0930
C     THE ABOVE QUANTITIES ARE USED IN COMPUTATIONS OF H MATRIX         MONB0940
      DEC=ASINF(DEX3*XREL(3))                                           MONB0942
      OUTPUT(5)=DEC/DR                                                  MONB0944
      RA=ATANF(XREL(2)/XREL(1))                                         MONB0946
      OUTPUT(3)=RA/DR                                                   MONB0948
      OUTPUT(1)=XMAG                                                    MONB0949
      DO 17 JJJ=1,4                                                     MONB0950
      IF(TYMEAS(JJJ,III))17,17,19                                       MONB0960
 19   CONTINUE                                                          MONB0962
      IF(JJJ-4)33,32,33                                                 MONB0964
 32   CONTINUE                                                          MONB0966
      DO 34 I=1,3                                                       MONB0968
      DO 34 J=1,3                                                       MONB0970
      STPARD(I,J)=0.                                                    MONB0972
      DO 34 K=1,3                                                       MONB0974
 34   STPARD(I,J)=STPARD(I,J)+DUM(K,I)*DUD(K,J)                         MONB0975
 33   CONTINUE                                                          MONB0976
      CALL COMPHQ(JJJ,2,NN)                                             MONB0978
 17   CONTINUE                                                          MONB0980
      WRITE OUTPUT TAPE NUTS,700,III,OUTPUT                             MONB0982
700   FORMAT(13H MOON BEACON ,I2,                                       MONB0983
     1/4H RNGE15.8,5H  RGRE15.8,                                        MONB0984
     25H   RAE15.8,5H  RARE15.8,5H  DECE15.8,5H  DCRE15.8)              MONB0985
      DO 20 I=1,6                                                       MONB0986
 20   OUTPUT(I)=0.                                                      MONB0987
 18   CONTINUE                                                          MONB0990
      RETURN                                                            MONB1000
      END                                                               MONB
```

Subroutine: MULT

Purpose: To find the product matrix of two 3 x 3 matrices.

Calling Sequence:

CALL MULT (A, B, C)

Input and **Output**

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	A	3,3			First input matrix
I	B	3,3			Second input matrix
O	C	3,3			C = AB

Common storages used or required: None

Subroutines required: None

Functions required: None

Approximate number of storages required:

MULT

```
*       LABEL                                                              MULT
*       SYMBOL TABLE                                                       MULT
        SUBROUTINE MULT(A,B,C,N)                                           MULT0000
        DIMENSION A(3,3),B(3,3),C(3,3)                                     MULT0010
        DO 1 I=1,3                                                         MULT0020
        DO 1 J=1,3                                                         MULT0030
        C(I,J) = 0.                                                        MULT0040
        DO 1 K=1,3                                                         MULT0050
      1 C(I,J) = C(I,J) + A(I,K)*B(K,J)                                    MULT0060
      3 RETURN                                                             MULT0070
        END                                                                MULT
```

Subroutine: NUTAIT

Purpose: To evaluate the elements of the nutation matrix which relates the Cartesian coordinates expressed in the true equator and equinox to those in the mean equator and equinox.

Calling Sequence:

CALL NUTAIT (TIME, OM, CR, DT, EN, EPSIL)

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	TIME	1	T	days	Total number days from ref.epoch
O	OM	1	Ω	radians	Argument of moon's descending
					node
O	CR	1	CR	radians	Mean longitude of moon
O	DT	1	$\Delta\Psi + d\Psi$	radians	Nutation in longitude of equinox
O	EN	3,3	N	(-)	Nutation matrix
O	EPSIL	1	$\bar{\epsilon} + d\epsilon$	radians	True obliquity

Common storages used or required: None

Subroutines required: None

Functions required: INTF, SINF, COSF

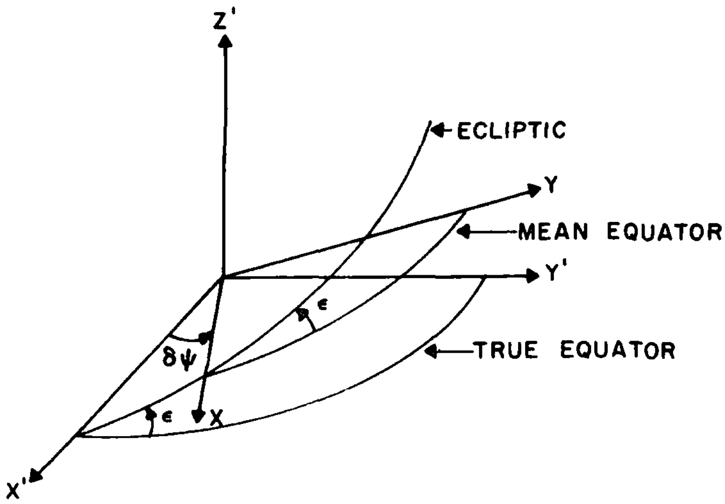
Approximate number of storages required:

NUTAIT

Method and Theory

The input time is converted to Julian centuries of 36525 days past the reference epoch (currently, 0^h January 1, 1950, E.T.). Where the computed angles are likely to be extremely large, the largest contributing term is reduced to the first revolution by the internal function DF(X). By using double precision arithmetic, this technique also helps to control round-off errors more closely as well as to keep the computed angles reasonable.

The relationship between the Cartesian coordinates expressed in the true equator and equinox and those expressed in the mean equator and equinox is shown in the following sketch:



The nutations are given by the following:

For obliquity $\delta\epsilon = \Delta\epsilon + d\epsilon$

For longitude $\delta\Psi = \Delta\Psi + d\Psi$

In the above equations, $\Delta\epsilon$, $\Delta\Psi$ express the long-period contributions and $d\epsilon$, $d\Psi$ the short-period contributions.

Both ϵ and Ψ are computed as functions of Ω, CR, Γ', L, Γ where

$$\Omega = 12.112790 - 360*DF(1) + .0020795T + .002081T^2 + .000002T^3$$

$$CR = 64.375452 + 360*DF(2) - .001131575T - .00113015T^2 - .0000019T^3$$

$$\Gamma' = 208.84399 + 360*DF(3) - .010334T - .010343T^2 - .000012T^3$$

$$\Gamma = 282.08053 + .0000470684D + .00045525T + .0004575T^2 + .000003T^3$$

$$L = 280.08121 + 360*DF(4) + .000303 (T + T^2)$$

T in the above equations is the number of Julian years past the reference epoch and the function DF(X) is

$$DF(X) = d \frac{a}{360}$$

where a = .052953922, 13.176397, .11140408, .98564734
and d is the number of days past the reference epoch.

Then

$$\begin{aligned}\Delta\epsilon \times 10^4 = {} & 25.5844 \cos \Omega - .2511 \cos 2 \Omega \\ & + 1.5336 \cos 2L + .0666 \cos (3L - \Gamma) \\ & - .0258 \cos (L + \Gamma) - .0183 \cos (2L - \Omega) \\ & - .0067 \cos (2\Gamma' - \Omega)\end{aligned}$$

$$\begin{aligned}d\epsilon \times 10^4 = {} & .2456 \cos 2CR + .0508 \cos (2CR - \Omega) \\ & + .0369 \cos (3CR - \Gamma') - .0139 \cos (CR + \Gamma') \\ & - .0086 \cos (CR - \Gamma' + \Omega) + .0083 \cos (CR - \Gamma' - \Omega) \\ & + .0061 \cos (3CR + \Gamma' - 2L) + .0064 \cos (3CR - \Gamma' - \Omega)\end{aligned}$$

$$\begin{aligned}\Delta\Psi \times 10^4 = {} & -(47.8927 + .0482T) \sin \Omega \\ & + .5800 \sin 2\Omega - 3.5361 \sin 2L - .1378 \sin (3L - \Gamma) \\ & + .0594 \sin (L + \Gamma) + .0344 \sin (2L - \Omega) + .0125 \sin (2\Gamma' - \Omega) \\ & + .3500 \sin (L - \Gamma) + .0125 \sin (2L - 2\Gamma')\end{aligned}$$

$$d\Psi \times 10^4 = -.5658 \sin 2CR - .0950 \sin (2CR - \Omega)$$
$$- .0725 \sin (3CR - \Gamma') + .0317 \sin (CR + \Gamma')$$
$$+ .0161 \sin (CR - \Gamma' + \Omega) + .0158 \sin (CR - \Gamma' - \Omega)$$
$$- .0144 \sin (3CR + \Gamma' - 2L) - .0122 \sin (3CR - \Gamma' - \Omega)$$
$$+ .1875 \sin (CR - \Gamma') + .0078 \sin (2CR - 2\Gamma')$$
$$+ .0414 \sin (CR + \Gamma' - 2L) + .0167 \sin (2CR - 2L)$$
$$- .0089 \sin (4CR - 2L).$$

The mean obliquity is calculated by

$$\bar{\epsilon} = 23.4457587 - .01309404T - .00000088T^2 + .00000050T^3$$

and the true obliquity by

$$\epsilon = \bar{\epsilon} + \delta\epsilon$$

The nutation matrix N relates the Cartesian coordinates expressed in the true equator and equinox to those in the mean equator and equinox by

$$\begin{vmatrix} X' \\ Y' \\ Z' \end{vmatrix} = N \begin{vmatrix} X \\ Y \\ Z \end{vmatrix}$$

where the primed system is the true equator and equinox and the unprimed is the mean equator and equinox.

Thus

$$N_{11} = \cos \delta\Psi$$
$$N_{12} = -\sin \delta\Psi \cos \bar{\epsilon}$$
$$N_{13} = \sin \delta\Psi \sin \bar{\epsilon}$$
$$N_{21} = \sin \delta\Psi \cos \epsilon$$

$$N_{22} = \cos\delta\Psi \cos\epsilon \cos\bar{\epsilon} + \sin\epsilon \sin\bar{\epsilon}$$

$$N_{23} = \cos\delta\Psi \cos\epsilon \sin\bar{\epsilon} - \sin\epsilon \cos\bar{\epsilon}$$

$$N_{31} = \sin\delta\Psi \sin\epsilon$$

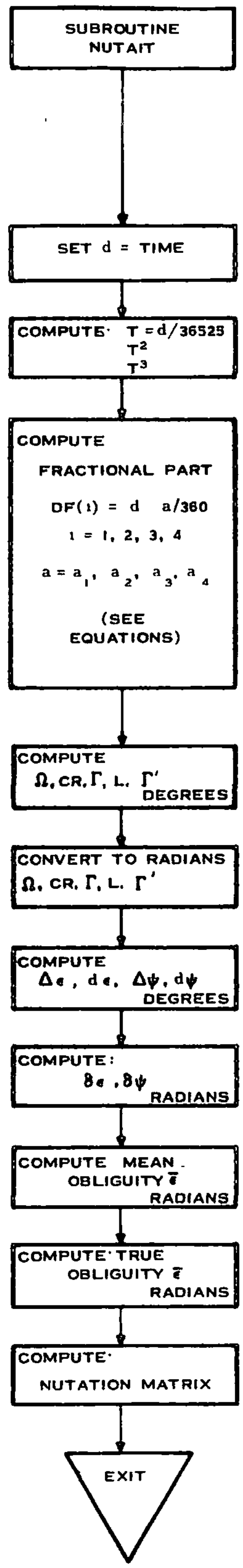
$$N_{32} = \cos\delta\Psi \sin\epsilon \cos\bar{\epsilon} - \cos\epsilon \sin\bar{\epsilon}$$

$$N_{33} = \cos\delta\Psi \sin\epsilon \sin\bar{\epsilon} + \cos\epsilon \cos\bar{\epsilon}$$

For numerical calculation, the above N matrix is expanded to the first order in $\delta\Psi$ and $\delta\epsilon$:

$$N = \begin{vmatrix} 1 & -\delta\Psi \cos\bar{\epsilon} & -\delta\Psi \sin\bar{\epsilon} \\ \delta\Psi \cos\bar{\epsilon} & 1 & -\delta\epsilon \\ \delta\Psi \sin\bar{\epsilon} & \delta\epsilon & 1 \end{vmatrix}$$

Reference: JPL Technical Report No. 32-223


SUBROUTINE NUTAIT
SET d = TIME
COMPUTE T = d/36525
T²
T³
COMPUTE
FRACTIONAL PART
DF(i) = d a/360
i = 1, 2, 3, 4
a = a1, a2, a3, a4
(SEE EQUATIONS)
COMPUTE
Ω, CR, Γ, L, Γ′
DEGREES
CONVERT TO RADIANS
Ω, CR, Γ, L, Γ′
COMPUTE
Δε, dε, Δψ, dψ
DEGREES
COMPUTE:
δε, δψ
RADIANS
COMPUTE MEAN OBLIGUITY ε̄
RADIANS
COMPUTE TRUE OBLIGUITY ε̃
RADIANS
COMPUTE
NUTATION MATRIX
EXIT

```
*     LABEL                                                              NUTA
*     SYMBOL TABLE                                                       NUTA
      SUBROUTINE NUTAIT (TIME,OM,CR,DT,EN,ERSIL)                         NUTA00
      DIMENSION EN(3,3)                                                  NUTA00
      D = TIME                                                           NUTA00
      T = D/36525.                                                       NUTA00
      T2 = T*T                                                           NUTA00
      T3 = T2*T                                                          NUTA00
      OM = 12.112790-.052953922*D+.0020795*T+.002081*T2+.000002*T3       NUTA00
      CR = 64.375452+13.176397*D-.001131575*T-.00113015*T2+.0000019*T3   NUTA00
      GP = 208.84399+.11140408*D-.010334*T-.010343*T2-.000012*T3         NUTA00
      VL = 280.08121+.98564734*D+.000303*(T+T2)                          NUTA00
      G = 282.08053+.0000470684*D+.00045525*T+.0004575*T2+.000003*T3     NUTA01
      OM = OM*.017453296                                                 NUTA01
      CR = CR*.017453296                                                 NUTA01
      GP = GP*.017453296                                                 NUTA01
      VL = VL*.017453296                                                 NUTA01
      G = G*.017453296                                                   NUTA01
      DE = 25.5844*COSF(OM)-.2511*COSF(2.*OM)+1.5336*COSF(2.*VL)         NUTA01
     1 +.0666*COSF(3.*VL-G)-.0258*COSF(VL+G)-.0183*COSF(2.*VL-OM)        NUTA01
     2 -.0067*COSF(2.*GP-OM)                                             NUTA01
      DD = .2456*COSF(2.*CR)+.0508*COSF(2.*CR-OM)+.0369*COSF(3.*CR-GP)   NUTA01
     1 -.0139*COSF(CR+GP)-.0086*COSF(CR-GP+OM)+.0083*COSF(CR-GP-OM)      NUTA02
     2 +.0061*COSF(3.*CR+GP-2.*VL)+.0064*COSF(3.*CR-GP-OM)               NUTA02
      DT = -(47.8927+.0482*T)*SINF(OM)+.58*SINF(2.*OM)                   NUTA02
     1 -3.5361*SINF(2.*VL)-.1378*SINF(3.*VL-G)+.0594*SINF(VL+G)          NUTA02
     2 +.0344*SINF(2.*VL-OM)+.0125*SINF(2.*GP-OM)+.35*SINF(VL-G)         NUTA02
     3 +.0125*SINF(2.*VL-2.*GP)                                          NUTA02
      DS = -.5658*SINF(2.*CR)-.095*SINF(2.*CR-OM)-.0725*SINF(3.*CR-GP)   NUTA02
     1 +.0317*SINF(CR+GP)+.0161*SINF(CR-GP+OM)+.0158*SINF(CR-GP-OM)      NUTA02
     2 -.0144*SINF(3.*CR+GP-2.*VL)-.0122*SINF(3.*CR-GP-OM)               NUTA02
     3 +.1875*SINF(CR-GP)+.0078*SINF(2.*CR-2.*GP)                        NUTA02
     4 +.0414*SINF(CR+GP-2.*VL)+.0167*SINF(2.*CR-2.*VL)                  NUTA03
     5 -.0089*SINF(4.*CR-2.*VL)                                          NUTA03
      DE = .17453296E-5*(DE+DD)                                          NUTA03
      DT = .17453296E-5*(DT+DS)                                          NUTA03
      EB = 23.4457587-.01309404*T-.00000088*T2+.0000005*T3               NUTA03
      EB = EB*.017453296                                                 NUTA03
      ERSIL = EB+DE                                                      NUTA03
      EN(1,1) = 1.                                                       NUTA03
      EN(1,2) = -DT*COSF(EB)                                             NUTA03
      EN(1,3) = -DT*SINF(EB)                                             NUTA03
      EN(2,1) = -EN(1,2)                                                 NUTA04
      EN(2,2) = 1.                                                       NUTA04
      EN(2,3) = -DE                                                      NUTA04
      EN(3,1) = -EN(1,3)                                                 NUTA04
      EN(3,2) = DE                                                       NUTA04
      EN(3,3) = 1.                                                       NUTA04
      RETURN                                                             NUTA04
      END                                                                NUTA
```

Page Intentionally Left Blank

Subroutine: OBLN

Purpose: To compute the perturbing acceleration due to the earth's oblateness. The formula for the earth's potential includes the second, third and fourth spherical harmonics. Part of the input is a rotation matrix to equator and equinox of date, obtained by successive calls to ROTEQ, NUTAIT, and MULT. It also computes the first variation of the perturbing acceleration for the variational equations.

Calling Sequence:

CALL OBLN (XP, U, A, VJ, H, D, PO, AN, B, NEQ)

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	XP	3	$\bar{R}$	Kilometers	Position vector
I	U	1	μ	Km^3/sec^2	Gravitational constant
I	A	1	R_E	Kilometers	Equatorial radius of the earth
I	VJ	1	J		Coefficient of second harmonic
I	H	1	H		Coefficient of third harmonic
I	D	1	D		Coefficient of fourth harmonic
O	PO	3	∇U		Perturbing acceleration
I	AN	3,3			Rotation matrix to equator and equinox of date

(Cont'd.)

Common storages used or required: None

Subroutines required: None

Functions required: SQRT

Approximate number of storages required: 480 DEC

(Cont'd.)

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
O	B	3,3	$\partial\ddot{X}_i/\partial X_j$		Derivatives of ∇U
I	NEQ	1			Number of differential equations
					being integrated

OBLN computes the perturbing acceleration due to the earth's oblateness. The oblate potential of the earth is assumed to contain the second, third, and fourth spherical harmonics:

$$U = \frac{\mu}{R}\left\{ \frac{JR_E^2}{3R^2}\,(1 - 3\sin^2\phi) + \frac{HR_E^3}{5R^3}\,(3\sin\phi - 5\sin^3\phi)\right.$$

$$\left. + \frac{DR_E^4}{35R^4}\,(3 - 30\sin^2\phi + 35\sin^4\phi)\right\}$$

where R_E is the equatorial radius of the earth and ϕ is the geocentric latitude, $\bar{R} = (X_1, X_2, X_3)$ is expressed in the mean equator and equinox of 1950.

The position vector $\bar{r} = (X, Y, Z)$ expressed in the true equator and equinox of date must be obtained to determine $\sin\phi$. The proper rotation matrix AN has been computed. Then,

$$\bar{r} = AN\,\bar{R}$$

and

$$\sin\phi = \frac{Z}{R}\,.$$

The perturbing acceleration is equal to ∇U:

$$\frac{\partial U}{\partial X_j} = - J\mu \frac{R_E^2}{R^4} \left\{ \left(1 - \frac{5Z^2}{R^2}\right) \frac{X_j}{R} + 2\,\frac{Z}{R}\, AN_{3j} \right\}$$

$$-H\mu \frac{R_E^3}{R^5} \left\{ \left(3 - \frac{7Z^2}{R^2}\right) \frac{Z}{R}\,\frac{X_j}{R} + \left(-\frac{3}{5} + \frac{3Z^2}{R^2}\right) AN_{3j} \right\}$$

$$-D\mu \frac{R_E^4}{R^6} \left\{ \left(\frac{3}{7} - \frac{6Z^2}{R^2} + \frac{9Z^4}{R^4}\right) \frac{X_j}{R} + \left(\frac{12}{7} - \frac{4Z^2}{R^2}\right) \frac{Z}{R}\, AN_{3j} \right\}$$

for $j = 1,2,3$.

If the number of second order differential equations is less than or equal to three, RETURN is called at this point.

OBLN also computes the derivatives of the perturbing acceleration for use in the integration package. For this purpose, only the term arising from the second harmonic is retained; also, the coordinates are regarded as being expressed in the reference system of the mean equator and equinox of 1950.

Starting with

$$\nabla U_a = \left(g_1\,\frac{X_1}{R},\; g_1\,\frac{X_2}{R},\; g_2\,\frac{X_3}{R}\right)$$

where

$$g_1 = -J\mu\,\frac{R_E^2}{R^4}\left(1 - \frac{5Z^2}{R^2}\right)$$

$$g_2 = -J\mu\,\frac{R_E^2}{R^4}\left(3 - \frac{5Z^2}{R^2}\right)$$

yields

$$\frac{\partial \ddot{X}_1}{\partial X_j} = g_1 \frac{X_1}{R} \left(\frac{1}{X_1} \frac{\partial X_1}{\partial X_j} - \frac{3}{R^2} \bar{R} \cdot \frac{\partial \bar{R}}{\partial X_j} \right)$$

$$+ J\mu R_E^2 \frac{X_1}{R^7} \left\{ 10X_3 \frac{\partial X_3}{\partial X_j} + 2 \left(1 - 10 \frac{X_3^2}{R^2} \right) \bar{R} \cdot \frac{\partial \bar{R}}{\partial X_j} \right\}$$

$$\frac{\partial \ddot{X}_2}{\partial X_j} = g_1 \frac{X_2}{R} \left(\frac{1}{X_2} \frac{\partial X_2}{\partial X_j} - \frac{3}{R^2} \bar{R} \cdot \frac{\partial \bar{R}}{\partial X_j} \right)$$

$$+ J\mu R_E^2 \frac{X_2}{R^7} \left\{ 10X_3 \frac{\partial X_3}{\partial X_j} + 2 \left(1 - 10 \frac{X_3^2}{R^2} \right) R \cdot \frac{\partial R}{\partial X_j} \right\}$$

$$\frac{\partial \ddot{X}_3}{\partial X_j} = g_2 \frac{X_3}{R} \left(\frac{1}{X_3} \frac{\partial X_3}{\partial X_j} - \frac{3}{R^2} \bar{R} \cdot \frac{\partial \bar{R}}{\partial X_j} \right)$$

$$+ J\mu R_E^2 \frac{X_3}{R^7} \left\{ 10X_3 \frac{\partial X_3}{\partial X_j} + 2 \left(3 - 10 \frac{X_3^2}{R^2} \right) R \cdot \frac{\partial \bar{R}}{\partial U_j} \right\}$$

Expanding the dot products and collecting terms gives

$$B_{11} = g_1 \left(\frac{1}{R} - \frac{3X_1^2}{R^3} \right) + 2J\mu R_E^2 \frac{X_1^2}{R^7} \left(1 - 10 \frac{X_3^2}{R^2} \right)$$

$$B_{12} = g_1 \left(- \frac{3X \cdot X_2}{R^3} \right) + 2 J\mu R_E^2 \frac{X_1 X_2}{R^7} \left(1 - 10 \frac{X_3^2}{R^2} \right)$$

$$B_{13} = g_1\left(-\frac{3X_1X_3}{R^3}\right) + 2\ J\mu\ R_E^2\ \frac{X_1X_3}{R^7}\left(6 - 10\frac{X_3^2}{R^2}\right)$$

$$B_{21} = B_{12}$$

$$B_{22} = g_1\left(\frac{1}{R} - \frac{3X_2^2}{R^3}\right) + 2\ J\mu\ R_E^2\ \frac{X_2^2}{R^7}\left(1 - 10\frac{X_3^2}{R^2}\right)$$

$$B_{23} = g_1\left(-\frac{3X_2X_3}{R^3}\right) + 2\ J\mu\ R_E^2\ \frac{X_2X_3}{R^7}\left(6 - 10\frac{X_3^2}{R^2}\right)$$

$$B_{31} = B_{13}$$

$$B_{32} = B_{23}$$

$$B_{33} = g_2\left(\frac{1}{R} - \frac{3X_3^2}{R^3}\right) + 2\ J\mu\ R_E^2\ \frac{X_3^2}{R^7}\left(8 - 10\frac{X_3^2}{R^2}\right)$$

where $\quad B\lambda_j = \frac{\partial \ddot{X}_i}{\partial X_j}$

For $R > 3R_E$, B is set equal to zero.

```
*       LABEL
*       SYMBOL TABLE
CEC2031  SUBROUTINE OBLN
        SUBROUTINE OBLN(XP,U,A,VJ,H,D,PO,AN,B ,NEQ)
        DIMENSION X(3),PO(3),AN(3,3),XP(3),B (3,3)
        RE=6378.15
        DO 2 I=1,3
        X(I) = 0.
        DO 2 J=1,3
      2 X(I) = X(I) + AN(I,J)*XP(J)
        R2 = X(1)**2+X(2)**2+X(3)**2
        R = SQRTF(R2)
        UR2 = U/R2
        AR = A/R
        AR2 = AR*AR
        AR3 = AR*AR2
        AR4 = AR*AR3
        ZR = X(3)/R
        ZR2 = ZR*ZR
        ZR4 = ZR2*ZR2
        TM1 = (1.-5.*ZR2)/R
        TM2 = 2.*ZR
        TM3 = (3.-7.*ZR2)*ZR/R
        TM4 = -.6+3.*ZR2
        TM5 = (.42857143-6.*ZR2+9.*ZR4)/R
        TM6 = (1.7142857-4.*ZR2)*ZR
        TO1 = -VJ*UR2*AR2
        TO2 = -H*UR2*AR3
        TO3 = -D*UR2*AR4
        DO 1 I=1,3
      1 PO(I) = TO1*(TM1*XP(I)+TM2*AN(3,I))
       1        +TO2*(TM3*XP(I)+TM4*AN(3,I))
       2        +TO3*(TM5*XP(I)+TM6*AN(3,I))
        IF(NEQ-3)3,3,4
      3 CONTINUE
        RETURN
      4 CONTINUE
        IF(R-3.*RE)5,5,6
      6 CONTINUE
        DO 7 I=1,3
        DO 7 J=1,3
        B(I,J)=0.
      7 CONTINUE
        GO TO 3
      5 CONTINUE
        G5=-TO1/R
        G6=5.*ZR2
        G1=-G5*(1.-G6)
        G2=-G5*(3. -G6)
        G7=2.*G6
        G8=X(1)/R
```

```
      G9=X(2)/R                                               OBLN048
      G10=X(3)/R                                              OBLN049
      B(1,1)=G1*(1.-3.*G8**2)+2.*G5*G8**2*(1.-G7)             OBLN050
      B(1,2)=-3.*G1*G8*G9+2.*G5*G8*G9*(1.-G7)                 OBLN051
      B(1,3)=-3.*G1*G8*G10+2.*G5*G8*G10*(6.-G7)               OBLN052
      B(2,1)=B(1,2)                                           OBLN053
      B(2,2)=G1*(1.-3.*G9**2)+2.*G5*G9**2*(1.-G7)             OBLN054
      B(2,3)=-3.*G1*G9*G10+2.*G5*G9*G10*(6.-G7)               OBLN055
      B(3,1)=B(1,3)                                           OBLN056
      B(3,2)=B(2,3)                                           OBLN057
      B(3,3)=G2*(1.-3.*G10**2)+2.*G5*G10**2*(8.-G7)           OBLN058
      GO TO 3                                                 OBLN059
      END                                                     OBLN
```

Page Intentionally Left Blank

Subroutine: ONBTR

Purpose: The subroutine obtains the relative position and velocity vectors from the vehicle to any of the six following bodies; Earth, Moon, Sun, Venus, Mars, and Jupiter. The bodies to be observed are established by input data. The types of observation which can be made on the body are range, range rate, right ascension, and declination. The selection of types of measurements to be made is also by input data. ONBTR calls subroutine COMPHO to perform the updating of the covariance matrix for observations being made.

Calling Sequence:

CALL ONBTR

Input and Output

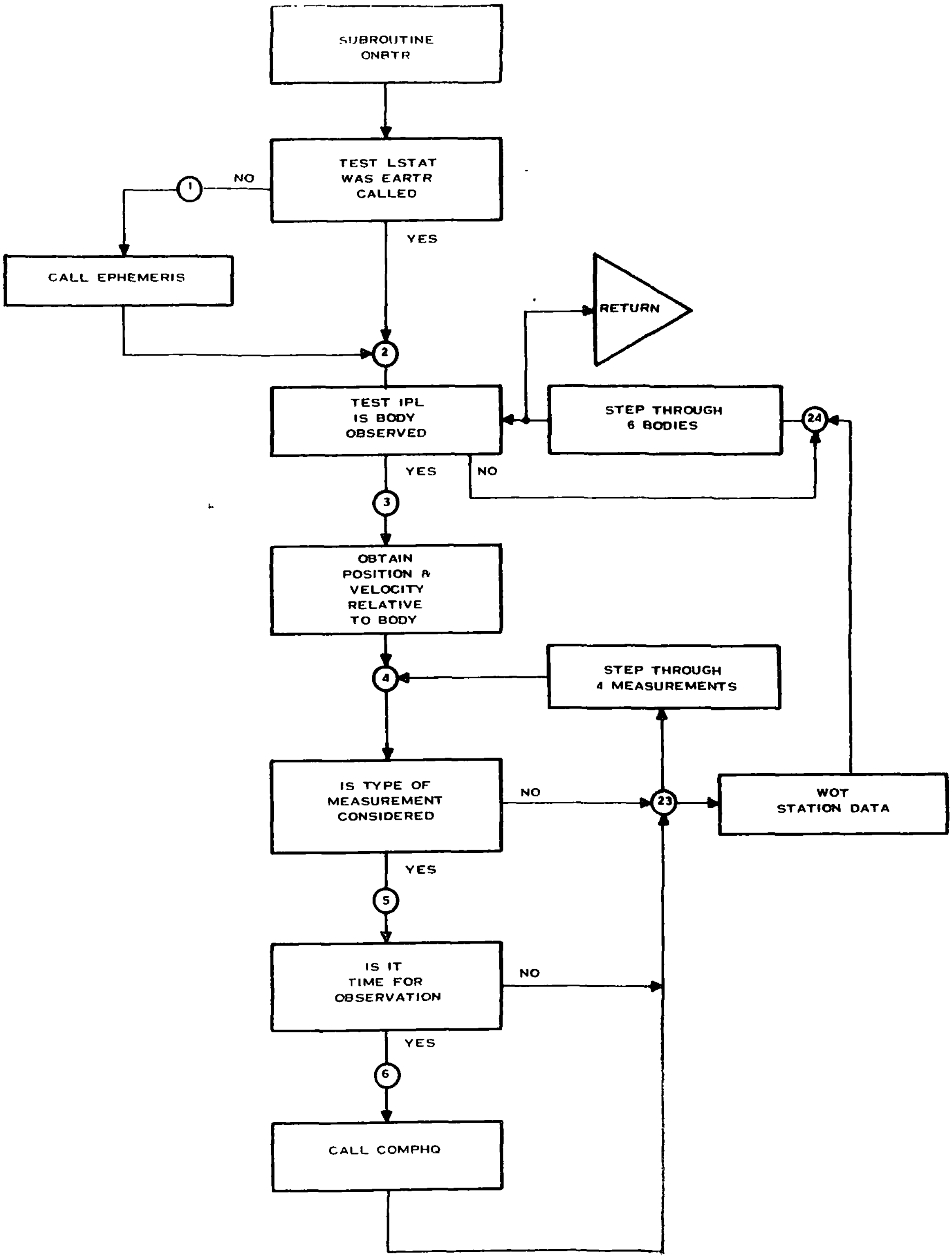
I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I					Quantities obtained from common
O					Quantities placed in common

Common storages used or required: T, S, C, IC

Subroutines required: COMPHO, INTRI

Functions required: ASIN, ATAN, FNORM, SQRT, (FIL)(SLO)(STH)

Approximate number of storages required: 267 DEC


SUBROUTINE
ONBTR
TEST LSTAT
WAS EARTR
CALLED
1
NO
CALL EPHEMERIS
YES
RETURN
2
TEST IPL
IS BODY
OBSERVED
STEP THROUGH
6 BODIES
24
YES
NO
3
OBTAIN
POSITION &
VELOCITY
RELATIVE
TO BODY
STEP THROUGH
4 MEASUREMENTS
4
IS TYPE OF
MEASUREMENT
CONSIDERED
NO
23
WOT
STATION DATA
YES
5
IS IT
TIME FOR
OBSERVATION
NO
YES
6
CALL COMPHQ

```
*     LABEL                                                          ONBT
*     SYMBOL TABLE                                                   ONBT
CEC2007                                                              ONBT
      SUBROUTINE ONBTR                                               ONBT000
C     SUBROUTINE ONBTR UPDATES CONVARIANCE MATRIX FOR ONBOARD        ONBT001
C     MEASUREMENTS OF RANGE,RIGHT ASCENSION,DECLINATION,AND          ONBT002
C     RANGE RATE RELATIVE TO THE 6 CELESTIAL BODIES                  ONBT003
      COMMON T,S,C,IC                                                ONBT004
      DIMENSION PO(22),VE(22),IMPLAN(4,6),IPL(6),X(3),VX(3)          ONBT005
     1,T(1360),S(1000),C(1000),IC(1),XREL(3),VREL(3)                 ONBT006
     2,TLAST(4,6),OBRATE(4,6),H(6),P(6,6),OUTPUT(6)                  ONBT007
      EQUIVALENCE (T,TDUM),(S,SDUM),(C,CDUM)                         ONBT008
     1,(IC(110),IPL),(IC(116),IMPLAN),(C(15),X),(C(18),VX)           ONBT009
     2,(IC(3),NOR),(C(13),TW),(C(14),TF),(IC(190),LSTAT)             ONBT010
     3,(C(649),TSECP),(C(30),TSEC),(S(437),OBRATE),(C(652),P)        ONBT011
     4,(C(622),TLAST),(C(62),PO),(C(84),VE)                          ONBT012
     1,(C(895),XREL),(C(898),VREL),(C(894), OBNO),(C(893),XMAG),     ONBT013
     2(C(892),DEX),(C(891),DEN1),(C(890),DEN2),(S(72),DR)            ONBT014
     3,(C(889),DEX2),(C(888),DEX3),(C(652),P),(C(973),OUTPUT)        ONBT015
      CALL SETN(NIN,NUTS)                                            ONBT015
      LSTAT=LSTAT                                                    ONBT016
      GO TO (1,2),LSTAT                                              ONBT017
    1 CONTINUE                                                       ONBT018
      NB=NOR-1                                                       ONBT019
      DIS=1.E10                                                      ONBT020
      CALL INTR1(TW,TF,NB,PO,1,VE,DIS)                               ONBT021
    2 CONTINUE                                                       ONBT022
      DO 24 JJ=1,6                                                   ONBT023
C     JJ DETERMINES BODY TO BE OBSERVED 1,EARTH 2,MOON 3,SUN         ONBT024
C     4,VENUS 5,MARS 6,JUPITER                                       ONBT025
      IF(IPL(JJ))24,24,3                                             ONBT026
    3 CONTINUE                                                       ONBT027
      LM=26-3*(JJ)                                                   ONBT028
      DO 4 I=1,3                                                     ONBT029
      MM=LM-I                                                        ONBT030
      XREL(I)=PO(MM)-X(I)                                            ONBT031
      VREL(I)=VE(MM)-VX(I)                                           ONBT032
    4 CONTINUE                                                       ONBT033
C     XREL=VECTOR FROM VEHICLE TO JJ-TH PLANET,EQUINOX 1950          ONBT034
C     VREL=RELATIVE VELOCITY VECTOR                                  ONBT035
      XMAG=FNORM(XREL)                                               ONBT036
      DEX=XREL(1)**2+XREL(2)**2                                      ONBT037
      DEN1=1./SQRTF(DEX)/XMAG**2                                     ONBT038
      DEN2=1./DEX                                                    ONBT039
      DEX2=XMAG*XMAG                                                 ONBT040
      DEX3=1./XMAG                                                   ONBT041
C     THE ABOVE QUANTITIES ARE USED IN COMPUTATIONS OF H MATRIX      ONBT042
      DELTT=TSEC-TSECP                                               ONBT043
      DEC=ASINF(DEX3*XREL(3))                                        ONBT043
      OUTPUT(5)=DEC/DR                                               ONBT043
      RA=ATANF(XREL(2)/XREL(1))                                      ONBT043
```

```
      OUTPUT(3)=RA/DR                                                   ONBT0438
      OUTPUT(1)=XMAG                                                    ONBT0439
      DO 23 JJJ=1,4                                                     ONBT0440
      IF(IMPLAN(JJJ,JJ))23,23,5                                         ONBT0450
    5 CONTINUE                                                          ONBT0460
      TLAST(JJJ,JJ)=TLAST(JJJ,JJ)+DELTT                                 ONBT0470
C     TLAST STORES TIME SINCE LAST JJJ TYPE OF OBSERVATION OF JJ BODY   ONBT0480
C     JJJ DETERMINES TYPE OF OBS.1=RANGE,2=RIGHT ASCENSION,             ONBT0490
C     3=DECLINATION,4=RANGE RATE                                        ONBT0500
      NOB=TLAST(JJJ,JJ)/OBRATE(JJJ,JJ)                                  ONBT0510
      OBNO=NOB                                                          ONBT0520
      IF(NOB)23,23,6                                                    ONBT0530
    6 CONTINUE                                                          ONBT0540
      JJJ=JJJ                                                           ONBT0550
      NN=426                                                            ONBT0560
      CALL COMPHQ(JJJ,1,NN)                                             ONBT0570
      TLAST(JJJ,JJ)=TLAST(JJJ,JJ)-OBNO*OBRATE(JJJ,JJ)                   ONBT0580
   23 CONTINUE                                                          ONBT0590
      WRITE OUTPUT TAPE NUTS,700,JJ,OUTPUT                              ONBT0591
  700 FORMAT(16H CELESTIAL BODY ,I2,                                    ONBT0592
     1/4H RNGE15.8,5H  RGRE15.8,                                        ONBT0593
     25H   RAE15.8,5H  RARE15.8,5H   DECE15.8,5H   DCRE15.8)            ONBT0594
      DO 20 I=1,6                                                       ONBT0595
   20 OUTPUT(I)=0.                                                      ONBT0596
   24 CONTINUE                                                          ONBT0600
      RETURN                                                            ONBT0610
      END                                                               ONBT
```

Subroutine: ORTC

Purpose: To convert Cartesian coordinates to conic elements for the output subroutines and to provide other auxiliary results used in computations in the output subroutine.

Calling Sequence:

CALL ORTC (X, DX, U, SMA, ECC, RCA, OINC, OMG, BEP, B, B2, R2, V2, A, C3, BET, THT)

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	X	3	$\vec{R}$	Km	Position vector
I	VX	3	$\vec{V}$	Km/sec	Velocity vector
I	U	1	μ	Km^3/sec^2	Central body gravitational
					constant
O	SMA	1	a	Km	Semi-Major Axis
O	ECC	1	e		Eccentricity
O	RCA	1	V_p	Km	Radius of closest approach
O	OINC	1	i	RAD	Orbital inclination

(Cont'd.)

Common storages used or required: ________

Subroutines required: ________

Functions required: ________

Approximate number of storages required: ________

(Cont'd.)

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
0	OMG	1	Ω	RAD	Argument of ascending mode
0	BEP	1	ω	RAD	Quantities used in output subroutine computations
0	B	3	$\vec{R} \times \vec{V}$		
0	B2	1	$\lvert\vec{R} \times \vec{V}\rvert^2$		
0	R2	1	$\lvert\vec{R}\rvert^2$	Km^2	
0	V2	1	$\lvert\vec{V}\rvert^2$	$(Km/sec)^2$	
0	A	1	$\vec{R} \cdot \vec{V}$		
0	C3	1	$\upsilon^2 - \frac{2\mu}{R}$	$(Km/sec)^2$	
0	BET	1	η	RAD	
0	THT	1	θ	RAD	

The orbital elements determined by the subroutine are the following:

$$C3 = \upsilon^2 - \frac{2\mu}{R} \qquad \vec{R} \times \vec{\upsilon} = \vec{h}$$

Semi-Major Axis

$$a = SMA = \frac{\mu}{|C3|}$$

Eccentricity

$$e = ECC \sqrt{1 + \frac{C3 \, |\vec{R} \times \vec{\upsilon}|^2}{\mu^2}} = \sqrt{1 + \frac{C_3 \, |\vec{h}|^2}{\mu^2}}$$

Radius of Closest Approach

$$V_p = RCA = \frac{|\vec{R} \times \vec{U}|^2}{\mu(1 + e)} = \frac{|\vec{h}|^2}{\mu(1 + e)}$$

Orbital Inclination

$$i = OINC = \mathrm{Tan}^{-1} \frac{\sqrt{h^2(1) + h^2(2)}}{h(3)}$$

Argument of the Ascending mode

$$\Omega = OMG = \mathrm{Tan}^{-1} \frac{h(1)}{-h(2)}$$

True Anomaly

$$\theta = THT = \mathrm{Tan}^{-1} \frac{(\vec{X} \cdot \vec{U})\ |\vec{h}|}{|h|^2 - \mu R}$$

Argument of Periapsis plus True Anomaly

$$n = BET = \frac{X(3)\ |\vec{h}|}{X(2)\ h(1) - X(1)\ h(2)}$$

Argument of Periapsis

$$\omega = BEP = BET - THT$$

```
*     LABEL
*     SYMBOL TABLE
      SUBROUTINE ORTC(X,DX,U,SMA,ECC,RCA,OINC,OMG,BEP
     1,B,B2,R2,V2,A,C3,BET,THT)
      DIMENSION X(3),DX(3),B(3)
      CALL CROSS(X,DX,B)
      R2 = DOT(X,X)
      V2 = DOT(DX,DX)
      B2 = DOT(B,B)
      A  = DOT(X,DX)
      BB = SQRTF(B2)
      R  = SQRTF(R2)
      C3 = V2-2.*U/R
      SMA= U/ABSF(C3)
      ECC= SQRTF(1.+C3*B2/U**2)
      OINC= ARKTNS(180,B(3),SQRTF(B(1)**2+B(2)**2))
      OMG= ARKTNS(360,-B(2),B(1))
      RCA= B2/(U*(1.+ECC))
      P  = B2/U
      THT= ARKTNS(360,U*(P-R),BB*A)
      BET= ARKTNS(360,X(2)*B(1)-X(1)*B(2),X(3)*BB)
      BEP= BET-THT
      IF(BEP) 2,3,3
    2 BEP=BEP+6.2831853
    3 CONTINUE
      RETURN
      END
```

Subroutine: OUTC

Purpose: OUTC writes out the trajectory elements in a coordinate system and reference system specified by an input quantity (KOUT). OUTC also writes out the orbital parameters found in ORTC, and writes out the selenographic latitude and longitude if the system is moon-centered.

Calling Sequence:

CALL OUTC

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
KOUT	IC(7)				0 for output in date
					1 for output in 1950
					2 for output in ecliptic coord.

Common storages used or required: T, S, C, IC

Subroutines required: see next page

Functions required: see next page

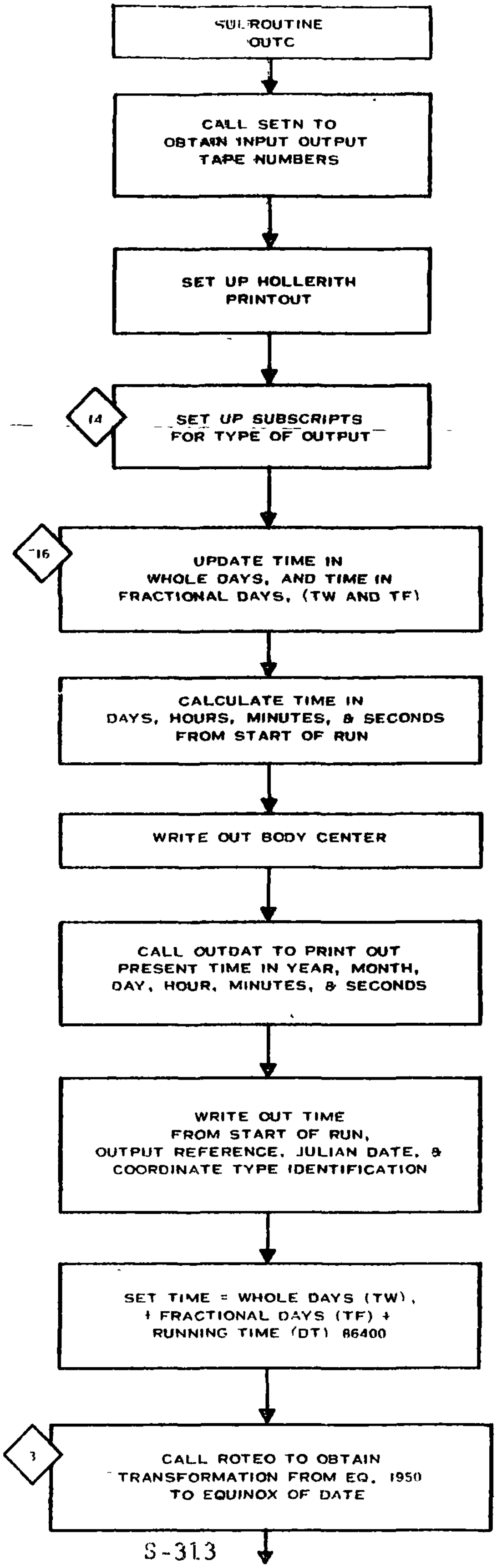
Approximate number of storages required: decimal 940; octal 1654

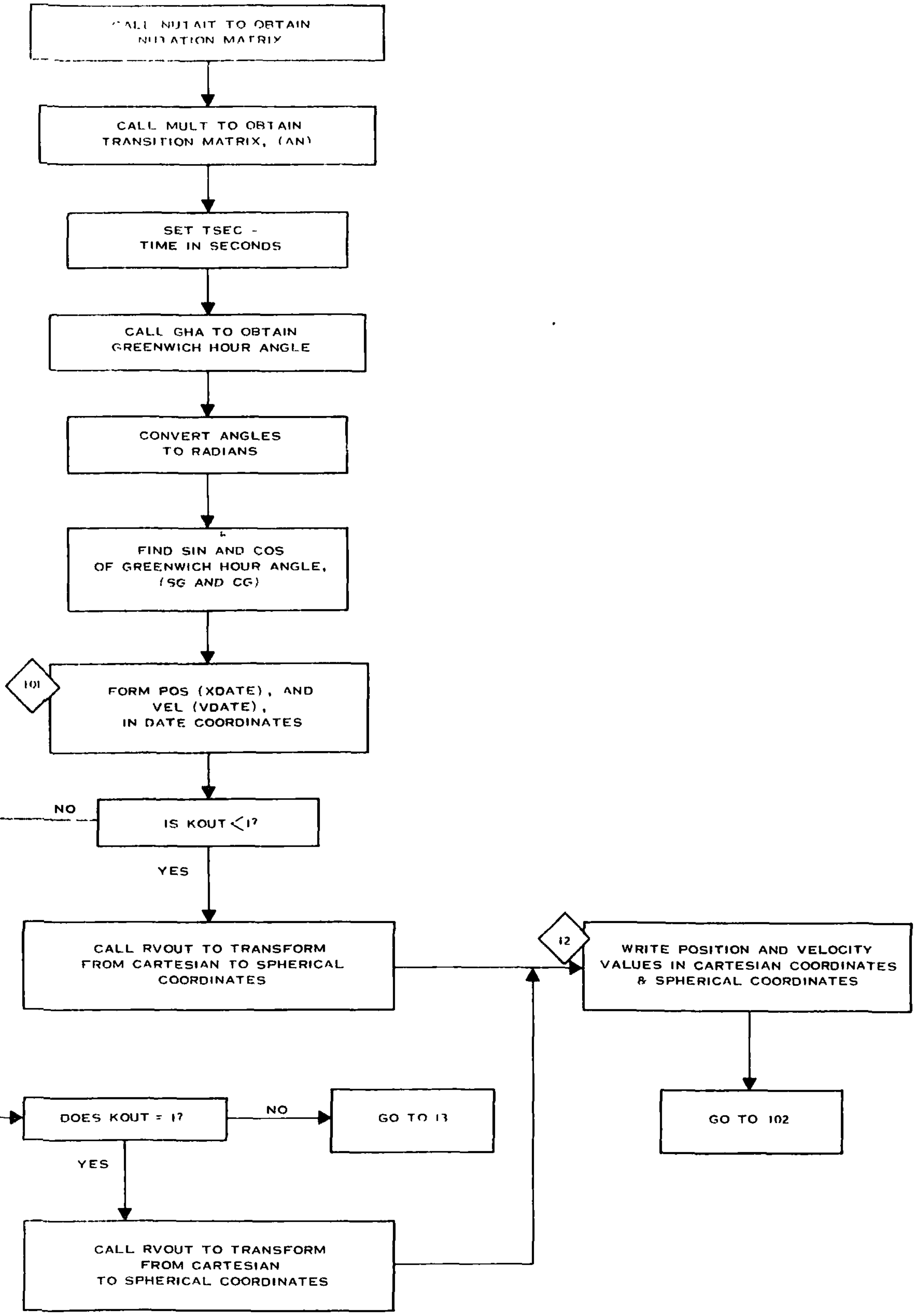
Subroutines Required

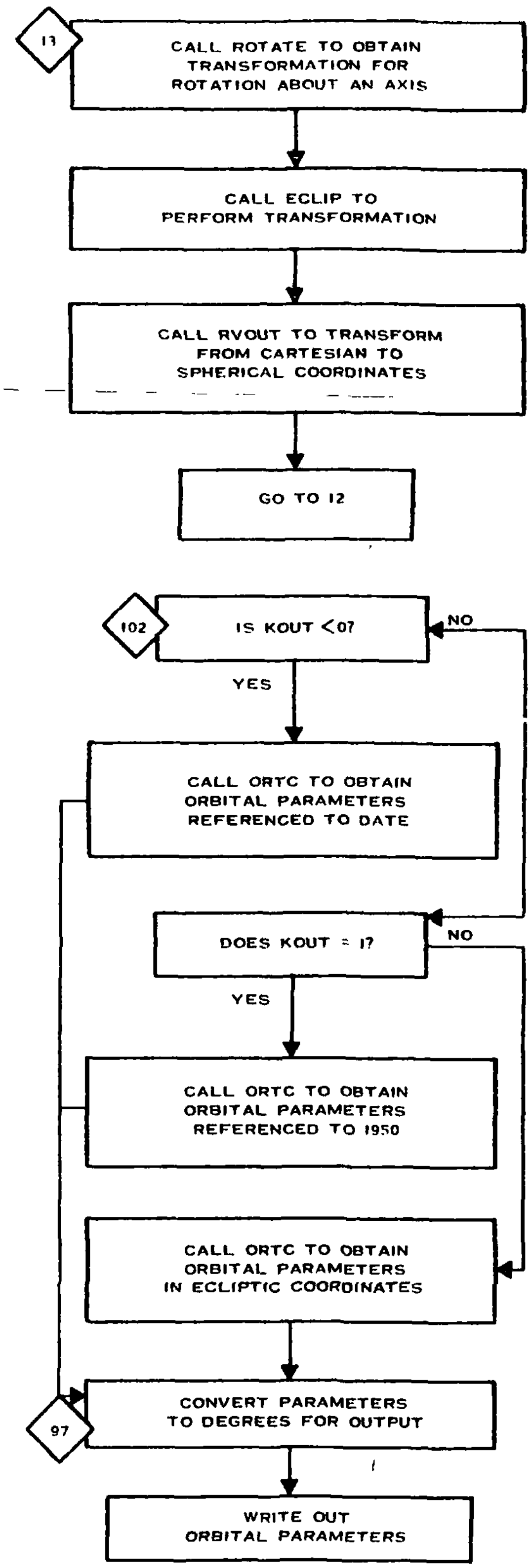
ECLIP
GHA
INTRI
MNAND
NMA
MULT
NUTAIT
ORTC
OUTDAT
ROTATE
ROTEQ
RVOUT
SETN

Functions Required

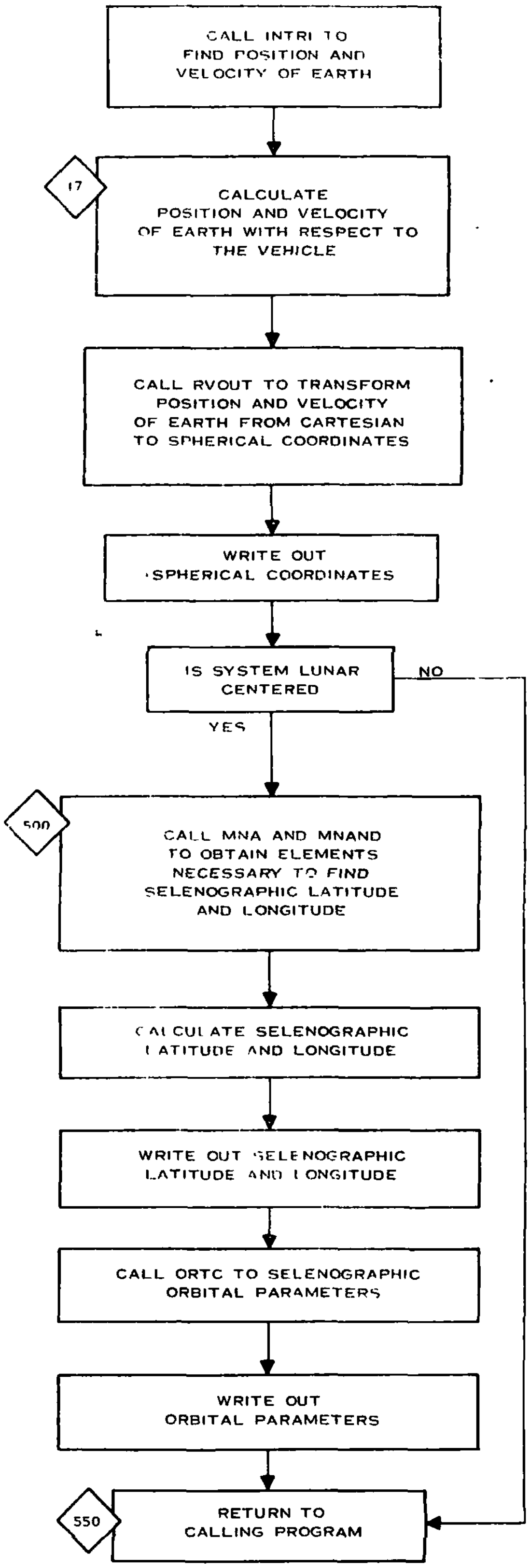
ARKTNS
ATAN
COS
SIN
SORT


SUBROUTINE
OUTC
CALL SETN TO
OBTAIN INPUT OUTPUT
TAPE NUMBERS
SET UP HOLLERITH
PRINTOUT
14
SET UP SUBSCRIPTS
FOR TYPE OF OUTPUT
16
UPDATE TIME IN
WHOLE DAYS, AND TIME IN
FRACTIONAL DAYS, (TW AND TF)
CALCULATE TIME IN
DAYS, HOURS, MINUTES, & SECONDS
FROM START OF RUN
WRITE OUT BODY CENTER
CALL OUTDAT TO PRINT OUT
PRESENT TIME IN YEAR, MONTH,
DAY, HOUR, MINUTES, & SECONDS
WRITE OUT TIME
FROM START OF RUN,
OUTPUT REFERENCE, JULIAN DATE, &
COORDINATE TYPE IDENTIFICATION
SET TIME = WHOLE DAYS (TW),
+ FRACTIONAL DAYS (TF) +
RUNNING TIME (DT) 86400
3
CALL ROTEO TO OBTAIN
TRANSFORMATION FROM EQ. 1950
TO EQUINOX OF DATE


CALL NUTAIT TO OBTAIN NUTATION MATRIX
CALL MULT TO OBTAIN TRANSITION MATRIX, (AN)
SET TSEC - TIME IN SECONDS
CALL GHA TO OBTAIN GREENWICH HOUR ANGLE
CONVERT ANGLES TO RADIANS
FIND SIN AND COS OF GREENWICH HOUR ANGLE, (SG AND CG)
101
FORM POS (XDATE), AND VEL (VDATE), IN DATE COORDINATES
NO
IS KOUT < 1?
YES
CALL RVOUT TO TRANSFORM FROM CARTESIAN TO SPHERICAL COORDINATES
12
WRITE POSITION AND VELOCITY VALUES IN CARTESIAN COORDINATES & SPHERICAL COORDINATES
DOES KOUT = 1?
NO
GO TO 13
GO TO 102
YES
CALL RVOUT TO TRANSFORM FROM CARTESIAN TO SPHERICAL COORDINATES


13
CALL ROTATE TO OBTAIN TRANSFORMATION FOR ROTATION ABOUT AN AXIS
CALL ECLIP TO PERFORM TRANSFORMATION
CALL RVOUT TO TRANSFORM FROM CARTESIAN TO SPHERICAL COORDINATES
GO TO 12
102
IS KOUT <07
NO
YES
CALL ORTC TO OBTAIN ORBITAL PARAMETERS REFERENCED TO DATE
DOES KOUT = 1?
NO
YES
CALL ORTC TO OBTAIN ORBITAL PARAMETERS REFERENCED TO 1950
CALL ORTC TO OBTAIN ORBITAL PARAMETERS IN ECLIPTIC COORDINATES
97
CONVERT PARAMETERS TO DEGREES FOR OUTPUT
WRITE OUT ORBITAL PARAMETERS

OUTC - 5


CALL INTRI TO FIND POSITION AND VELOCITY OF EARTH
17
CALCULATE POSITION AND VELOCITY OF EARTH WITH RESPECT TO THE VEHICLE
CALL RVOUT TO TRANSFORM POSITION AND VELOCITY OF EARTH FROM CARTESIAN TO SPHERICAL COORDINATES
WRITE OUT SPHERICAL COORDINATES
IS SYSTEM LUNAR CENTERED
NO
YES
500
CALL MNA AND MNAND TO OBTAIN ELEMENTS NECESSARY TO FIND SELENOGRAPHIC LATITUDE AND LONGITUDE
CALCULATE SELENOGRAPHIC LATITUDE AND LONGITUDE
WRITE OUT SELENOGRAPHIC LATITUDE AND LONGITUDE
CALL ORTC TO SELENOGRAPHIC ORBITAL PARAMETERS
WRITE OUT ORBITAL PARAMETERS
550
RETURN TO CALLING PROGRAM

```
*      LABEL                                                           OUTC
*      SYMBOL TABLE                                                    OUTC
CEC2015  SUBROUTINE OUTC                                               OUTC
       SUBROUTINE OUTC                                                 OUTC000
       COMMON T,S,C,IC                                                 OUTC001
       EQUIVALENCE (T,TDUM),(S,SDUM),(C,CDUM),(IC,ICDUM)               OUTC002
       EQUIVALENCE (T(2),DT),(C(15),XP),(C(18),VXP),(C(129),EA)        OUTC003
      1,(C(120),EN),(C(138),AN),(C(62),PO),(C(84),VE),(C(10),TW)       OUTC004
      2,(C(11),TF),(IC(3),NOR),(IC(6),ITARG),(IC(7),KOUT),(C(55),U)    OUTC005
      3,(S(73),RTD),(C(180),SMA),(C(181),ECC),(C(182),RCA),(C(183),OINC) OUTC006
      4,(C(184),OMG),(C(185),BEP),(C(186),XED),(C(189),VED)            OUTC007
      5,(C(156),ECX),(C(159),ECV),(C(147),ECL),(C(30),T2)              OUTC008
      6,(C(192),RCV),(C(195),RCV2),(C(196),R2),(C(197),V2),(C(198),RDV) OUTC009
      7,(C(199),C3),(C(162),BET),(C(163),THT)                          OUTC010
       DIMENSION T(1360),S(1000),C(1000),IC(1)                         OUTC011
       DIMENSION XP(3),VXP(3),EA(3,3),EN(3,3),B(3,3),AN(3,3),          OUTC012
      1 BAN(3,3),PO(22),VE(22),XDATE(3),VDATE(3),XEARTH(3),VEARTH(3),  OUTC013
      2SPDATE(6),ECX(3),ECV(3),CNAME(6),VEV(3)                         OUTC014
       DIMENSION XED(3),VED(3),PEV(3),RCV(3),RCVM(3)                   OUTC015
       DIMENSION ANAME(2)                                              OUTC016
       DIMENSION BNAME(4)                                              OUTC017
       DIMENSION ECL(3,3)                                              OUTC018
       DIMENSION POS(3),VES(3),POM(3),VEM(3),POT(3),VET(3)             OUTC019
       CALL SETN(NIN,NOUT)                                             OUTC019
       ANAME(1) = 6H  DATE                                             OUTC020
       ANAME(2) = 6H1950.0                                             OUTC021
       BNAME(1) = 6H    EC                                             OUTC022
       BNAME(2) = 6HLIPTIC                                             OUTC023
       BNAME(3) = 6H  EQUA                                             OUTC024
       BNAME(4) = 6HTORIAL                                             OUTC025
       CNAME(1)=6H EARTH                                               OUTC026
       CNAME(2)=6H  MOON                                               OUTC027
       CNAME(3)=6H   SUN                                               OUTC028
       CNAME(4)=6H VENUS                                               OUTC029
       CNAME(5)=6H  MARS                                               OUTC030
       CNAME(6)=6HJUPITR                                               OUTC031
  202  CONTINUE                                                        OUTC032
       KO=KOUT                                                         OUTC033
       NOR = NOR                                                       OUTC034
       IF (KOUT-1) 14,14,15                                            OUTC035
    14 KT = 3                                                          OUTC036
       GO TO 16                                                        OUTC037
    15 KT = 1                                                          OUTC038
       KO=0                                                            OUTC039
    16 CONTINUE                                                        OUTC040
       TP2 = TF+DT/86400.                                              OUTC041
       TWGN=0.                                                         OUTC042
    32 CONTINUE                                                        OUTC043
       IF(TP2) 33,34,34                                                OUTC044
    33 CONTINUE                                                        OUTC045
       TP2=TP2+1.                                                      OUTC046
```

```
      TWGN=TWGN-1.                                                      OUTC0470
      GO TO 32                                                          OUTC0480
   34 CONTINUE                                                          OUTC0490
      TFP = INTF(TP2)                                                   OUTC0500
      TP2 = TP2-TFP                                                     OUTC0510
      TP1=TW+TFP+TWGN                                                   OUTC0520
      IT1=T2/86400.                                                     OUTC0530
      DT1=IT1                                                           OUTC0540
      DT1=T2-DT1*86400.                                                 OUTC0550
      IT2=DT1/3600.                                                     OUTC0560
      DT2=IT2                                                           OUTC0570
      DT2=DT1-DT2*3600.                                                 OUTC0580
      IT3=DT2/60.                                                       OUTC0590
      DT3=IT3                                                           OUTC0600
      DT3=DT2-DT3*60.                                                   OUTC0610
      TT = INTF(TP2+.5)                                                 OUTC0620
      TP2P = TP2+.5-TT                                                  OUTC0630
      TP1P = TP1 + TT + 2433282.                                        OUTC0640
      WRITE OUTPUT TAPE NOUT,702,CNAME(NOR)                             OUTC0650
702   FORMAT (//1H0A6,10H  CENTERED)                                    OUTC0660
      CALL OUTDAT(TP1,TP2)                                              OUTC0670
      WRITE OUTPUT TAPE NOUT,2,IT1,IT2,IT3,DT3,ANAME(KO  +1),TP1P,TP2P  OUTC0680
     1 ,BNAME(KT),BNAME(KT+1)                                           OUTC0690
    2 FORMAT( I4,5H DAYSI4,5H HRS.I4,5H MIN.F7.2,5H SEC.4XA6,3X12HJULIANOUTC0700
     1 DATE F8.0,F8.8,10X2A6,12H COORDINATES)                           OUTC0710
      N = NOR                                                           OUTC0720
      TIME = TW+TF+DT/86400.                                            OUTC0730
      DIS = 1.E10                                                       OUTC0740
      NR1 = N-1                                                         OUTC0750
      NV = 1                                                            OUTC0760
    3 CALL ROTEQ(TIME,EA)                                               OUTC0770
      CALL NUTAIT(TIME,WM,CR,DA,EN,EPSIL)                               OUTC0780
      CALL MULT(EN,EA,AN,0)                                             OUTC0790
      TSEC = 86400.*TP2                                                 OUTC0800
      CALL GHA(TSEC,TP1,GHAN,EN(2,1),OMEGA)                             OUTC0810
      GHAN = GHAN*.017453296                                            OUTC0820
      OMEGA = OMEGA*.017453296                                          OUTC0830
      SG = SINF(GHAN)                                                   OUTC0840
      CG = COSF(GHAN)                                                   OUTC0850
      DO 1 I=1,3                                                        OUTC0860
      XDATE(I) = 0.                                                     OUTC0870
      VDATE(I) = 0.                                                     OUTC0880
      DO 101 J=1,3                                                      OUTC0890
      XDATE(I) = XDATE(I) + AN(I,J)*XP(J)                               OUTC0900
      VDATE(I) = VDATE(I) + AN(I,J)*VXP(J)                              OUTC0910
101   CONTINUE                                                          OUTC0920
      XED(I)=XDATE(I)                                                   OUTC0930
      VED(I)=VDATE(I)                                                   OUTC0940
    1 CONTINUE                                                          OUTC0950
      IF (KOUT-1) 11,10,13                                              OUTC0960
10    CALL RVOUT(XP,VXP,SPDATE)                                         OUTC0970
```

```
      WRITE OUTPUT TAPE NOUT,5,XP,VXP,SPDAT                             OUTC098
      GO TO 102                                                          OUTC099
 11   CALL RVOUT(XDATE,VDATE,SPDATE)                                     OUTC100
 12   WRITE OUTPUT TAPE NOUT,5,XDATE,VDATE,SPDATE                        OUTC101
      GO TO 102                                                          OUTC102
 13   CALL ROTATE(1,EPSIL,ECL,1)                                         OUTC103
      CALL ECLIP(XDATE,VDATE,ECL)                                        OUTC104
      CALL RVOUT(XDATE,VDATE,SPDATE)                                     OUTC105
      DO 103 I=1,3                                                       OUTC106
      ECX(I)=XDATE(I)                                                    OUTC107
103   ECV(I)=VDATE(I)                                                    OUTC108
      GO TO 12                                                           OUTC109
102   CONTINUE                                                           OUTC110
  5   FORMAT(4H   XE15.8,5H    YE15.8,5H    ZE15.8,5H   DXE15.8,         OUTC111
     1 5H   DYE15.8,5H   DZE15.8/4H    RE15.8,5H  DECE15.8,5H    RAE15.8, OUTC112
     2 5H    VE15.8,5H  PTHE15.8,5H    AZE15.8)                          OUTC113
      IF (KOUT-1) 105,106,107                                            OUTC114
105   CALL ORTC(XED,VED,U,SMA,ECC,RCA,OINC,OMG,BEP                       OUTC115
     1,RCV,RCV2,R2,V2,RDV,C3,BET,THT)                                    OUTC116
      GO TO 97                                                           OUTC117
106   CALL ORTC(XP,VXP,U, SMA,ECC,RCA,OINC,OMG,BEP                       OUTC118
     1,RCV,RCV2,R2,V2,RDV,C3,BET,THT)                                    OUTC119
      GO TO 97                                                           OUTC120
107   CALL ORTC(ECX,ECV,U,SMA,ECC,RCA,OINC,OMG,BEP                       OUTC121
     1,RCV,RCV2,R2,V2,RDV,C3,BET,THT)                                    OUTC122
 97   CONTINUE                                                           OUTC123
      ORIN =OINC*RTD                                                     OUTC124
      OMEG =OMG*RTD                                                      OUTC125
      APF =BEP*RTD                                                       OUTC126
      WRITE OUTPUT TAPE NOUT,4,SMA,ECC,ORIN,OMEG,APF,RCA                 OUTC127
      CALL INTR1(TP1,TP2,NR1,PO,NV,VE,DIS)                               OUTC128
      DO 17 I=1,3                                                        OUTC129
      I4=23-I                                                            OUTC130
      PEV(I)=PO(I4)-XP(I)                                                OUTC131
      VEV(I)=VXP(I)-VE(I4)                                               OUTC132
 17   CONTINUE                                                           OUTC133
      DO 108 I=1,3                                                       OUTC134
      XDATE(I)=0.                                                        OUTC135
      VDATE(I)=0.                                                        OUTC136
      DO 108 J=1,3                                                       OUTC137
      XDATE(I)=XDATE(I)-AN(I,J)*PEV(J)                                   OUTC138
108   VDATE(I)=VDATE(I)+AN(I,J)*VEV(J)                                   OUTC139
 22   XEARTH(3) = XDATE(3)                                               OUTC140
      VEARTH(3) = VDATE(3)                                               OUTC141
      XEARTH(1) = XDATE(1)*CG + XDATE(2)*SG                              OUTC142
      XEARTH(2) = -XDATE(1)*SG + XDATE(2)*CG                             OUTC143
      VEARTH(1)=VDATE(1)*CG+VDATE(2)*SG                                  OUTC144
      VEARTH(2)=-VDATE(1)*SG+VDATE(2)*CG                                 OUTC145
      CALL RVOUT(XEARTH,VEARTH,SPDATE)                                   OUTC146
      WRITE OUTPUT TAPE NOUT,31,SPDATE                                   OUTC147
 31   FORMAT(4H RTEE15.8,5H  LATE15.8,                                   OUTC148
```

```
   3 5H   LONE15.8,5H    VEE15.8,5H   PTEE15.8,5H   AZEE15.8)          OUTC1490
 29  CONTINUE                                                          OUTC1500
     IF(N-2)550,500,550                                                OUTC1510
500  CALL MNA(TIME,WM,CR,DA,EPSIL,RR,G,GP,WW,BAN)                      OUTC1520
     DO 510 I=1,3                                                      OUTC1540
     XDATE(I)=0.                                                       OUTC1550
     VDATE(I)=0.                                                       OUTC1560
     DO 510 J=1,3                                                      OUTC1570
     XDATE(I)=XDATE(I)+BAN(I,J)*XED(J)                                 OUTC1580
510  VDATE(I)=VDATE(I)+BAN(I,J)*VED(J)                                 OUTC1590
     SLAT=XDATE(3)/SQRTF(XDATE(1)**2+XDATE(2)**2)                      OUTC1600
     SLAT=ATANF(SLAT)*RTD                                              OUTC1610
     SLNG=ARKTNS(360,XDATE(1),XDATE(2))*RTD                            OUTC1620
     WRITE OUTPUT TAPE NOUT,701,SLNG,SLAT                              OUTC1630
701  FORMAT(19H SELENOGRAPHIC LON=,E17.8,5H LAT=,E17.8)                OUTC1640
     CALL ORTC(XDATE,VDATE,U,AMS,EPS,RP,ORIN,OMEG,APF                  OUTC1650
    1,RCVM,RCVM2,R2M,V2M,RDVM,C3M,BETM,THTM)                           OUTC1660
     OMEG = OMEG*RTD                                                   OUTC1670
     ORIN = ORIN*RTD                                                   OUTC1680
C    GIVES CALENDER DATE FROM T1(WHOLE DAYS FROM 1950)                 OUTC1690
C    ANDT2(FRACT OF DAY)                                               OUTC1700
     APF = APF*RTD                                                     OUTC1710
     WRITE OUTPUT TAPE NOUT,4,AMS,EPS,ORIN,OMEG,APF,RP                 OUTC1720
   4 FORMAT (4H SMAE15.8,5H   ECCE15.8,5H   INCE15.8,5H   LANE15.8,    OUTC1730
    1 5H   APFE15.8,5H   RCAE15.8)                                     OUTC1740
 550 RETURN                                                            OUTC1750
     END                                                               OUTC
```

Subroutine: OUTDAT

Purpose: OUTDAT finds and outputs the calendar date and Julian date, given the number of days since 1950.0.

Calling Sequence:

CALL OUTDAT (T1, T2)

Input and Output

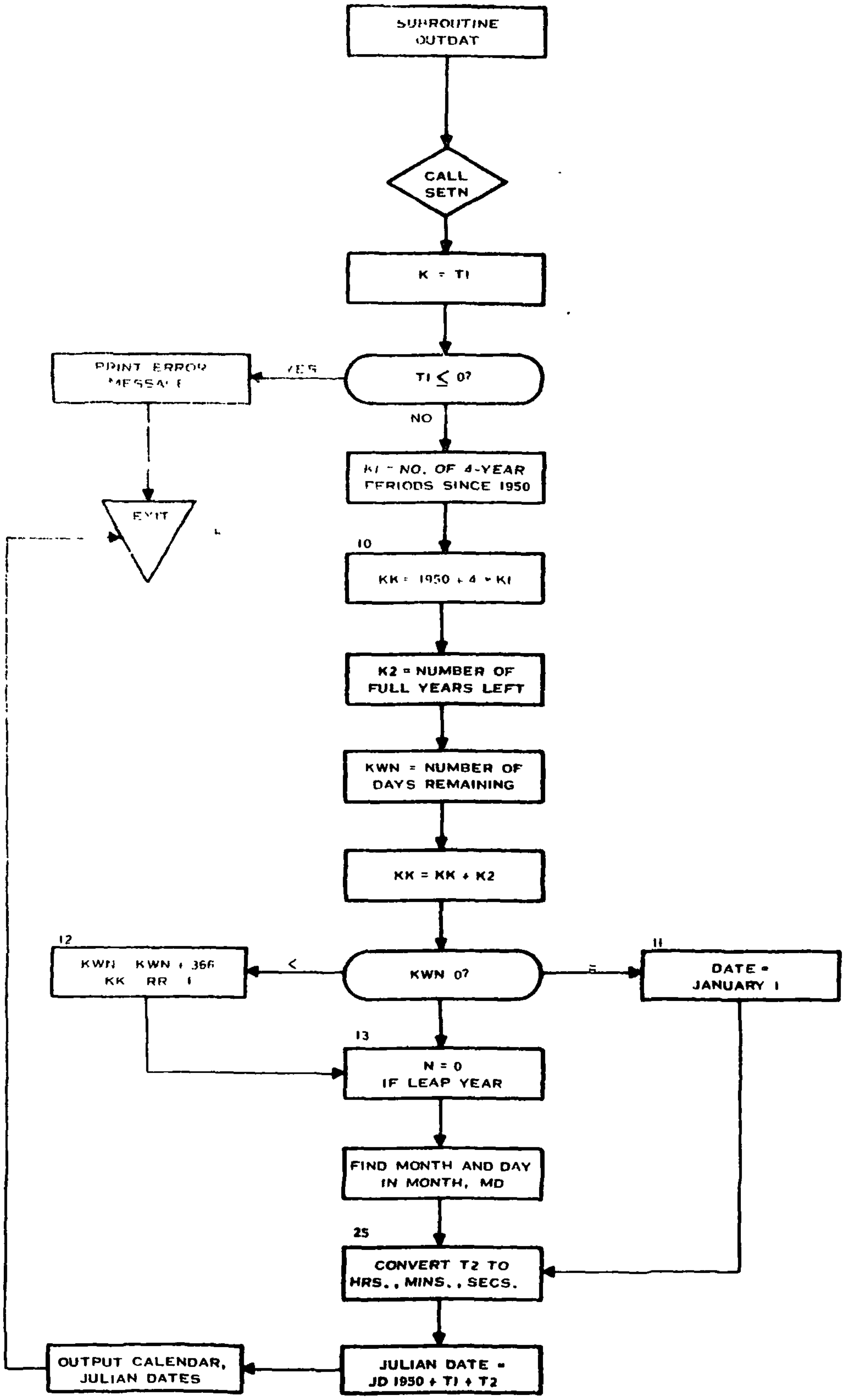
I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	T1	1		days	Integral number of days from
					1950.0.
I	T2	1		days	Fraction of day past T1

Common storages used or required: None

Subroutines required: SETN

Functions required: XMODF, INTF

Approximate number of storages required: 351 DEC


SUBROUTINE OUTDAT
CALL SETN
K = TI
TI ≤ 0?
YES
PRINT ERROR MESSAGE
EXIT
NO
KI = NO. OF 4-YEAR PERIODS SINCE 1950
10
KK = 1950 + 4 * KI
K2 = NUMBER OF FULL YEARS LEFT
KWN = NUMBER OF DAYS REMAINING
KK = KK + K2
KWN 0?
<
12
KWN KWN + 366
KK RR 1
=
11
DATE = JANUARY 1
13
N = 0 IF LEAP YEAR
FIND MONTH AND DAY IN MONTH, MD
25
CONVERT T2 TO HRS., MINS., SECS.
JULIAN DATE = JD 1950 + TI + T2
OUTPUT CALENDAR, JULIAN DATES

```
*      LABEL
*      SYMBOL TABLE
CEC20DT
       SUBROUTINE OUTDAT (T1,T2)
C      GIVES CALENDER DATE FROM T1(WHOLE DAYS FROM 1950)
C      ANDT2(FRACT OF DAY)
       CALL SETN(NIN,NOUT)
       MD=0
       K=T1
       IF(T1)6,6,7
  6    WRITE OUTPUT TAPE NOUT,100
  8    RETURN
 100   FORMAT(23H DATE IS 1950 OR BEFORE)
  7    K1=K/1461
 10    KK=K1*4+1950
       K1=XMODF(K,1461)
       K2=K1/365
       KWN=XMODF(K1,365)-K2/3
       KK=KK+K2
       IF(KWN)12,11,13
 11    MONTH=1
       MD=1
       GO TO 25
 12    KWN=KWN+366
       KK=KK-1
 13    N=XMODF(KK,4)
       KNW=0
       JJ=0
 135   JJ=JJ+1
       IF(12-JJ)23,23,137
C      J,F,M,A,M,J,J,A,S,O,N,MONTHS FOR GO TO
 137   GO TO (14,15,14,16,14,16,14,14,16,14,16,14),JJ
 14    CONTINUE
       KNW=KNW+31
       GO TO 17
 15    CONTINUE
       IF(N)18,18,19
 18    KNW=KNW+1
 19    KNW=KNW+28
       GO TO 17
 16    KNW=KNW+30
 17    CONTINUE
       IF(KWN-KNW)20,21,22
 22    CONTINUE
       MD=KNW
       GO TO 135
    20 CONTINUE
       MONTH=JJ
       MD=KWN-MD+1
       GO TO 25
```

```
 21 CONTINUE
      MONTH=JJ+1
     MD=1
     GO TO 25
 23  MONTH =12
     MD=KWN-MD+1
 25  CONTINUE
     TH=T2*24.
      THP=INTF(TH)
     NHOUR=THP
      THP=(TH-THP)*60.
     TH=INTF(THP)
      NMIN=TH
     THP=(THP-TH)*60.
     NSEC=THP
     TSEC=NSEC
     THP=(THP-TSEC)*1000.
     NFSEC=THP
     TT=INTF(T2+.5)
     TP2=T2+.5-TT
     TP1=T1+TT+2433282.
     WRITE OUTPUT TAPE NOUT,101,KK,MONTH,MD,NHOUR,NMIN,NSEC,NFSEC
    1,TP1,TP2
101 FORMAT(6H YEAR=,I4,8H  MONTH=,I2, 6H  DAY=,I2, 7H  HOUR=,I2,
    1 6H  MIN=,I2,6H  SEC=,I2,1H.,I3,10X,12HJULIAN DATE=,
    2F8.0,F8.8)
     GO TO 8
     END
```

Subroutine: OUTP

Purpose: Obtain the error data in the desired coordinate system for output purposes. The three coordinate systems available are: (1) Equator of Date, (2) Equator of 1950, and (3) Ecliptic. The coordinate system is determined by an input quantity KOUT.

Calling Sequence:

CALL OUTP

Input and **Output**

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I					Quantities obtained from common
O					Quantities printed out

Common storages used or required: T, S, C, IC

Subroutines required: CROSS, HPHT, INV3, MATRX, PTRAN

Functions required: COS, FNORM, SIN, SQRT

Approximate number of storages required: 1179 DEC

A. <u>Derivation of Partials Used in Subroutine</u>

In order to determine the RMS error in knowledge of orbital elements, the partials which relate the elements and the vehicle state are required. The derivation of these partials follows. The angular elements are shown below in Fig. 1.

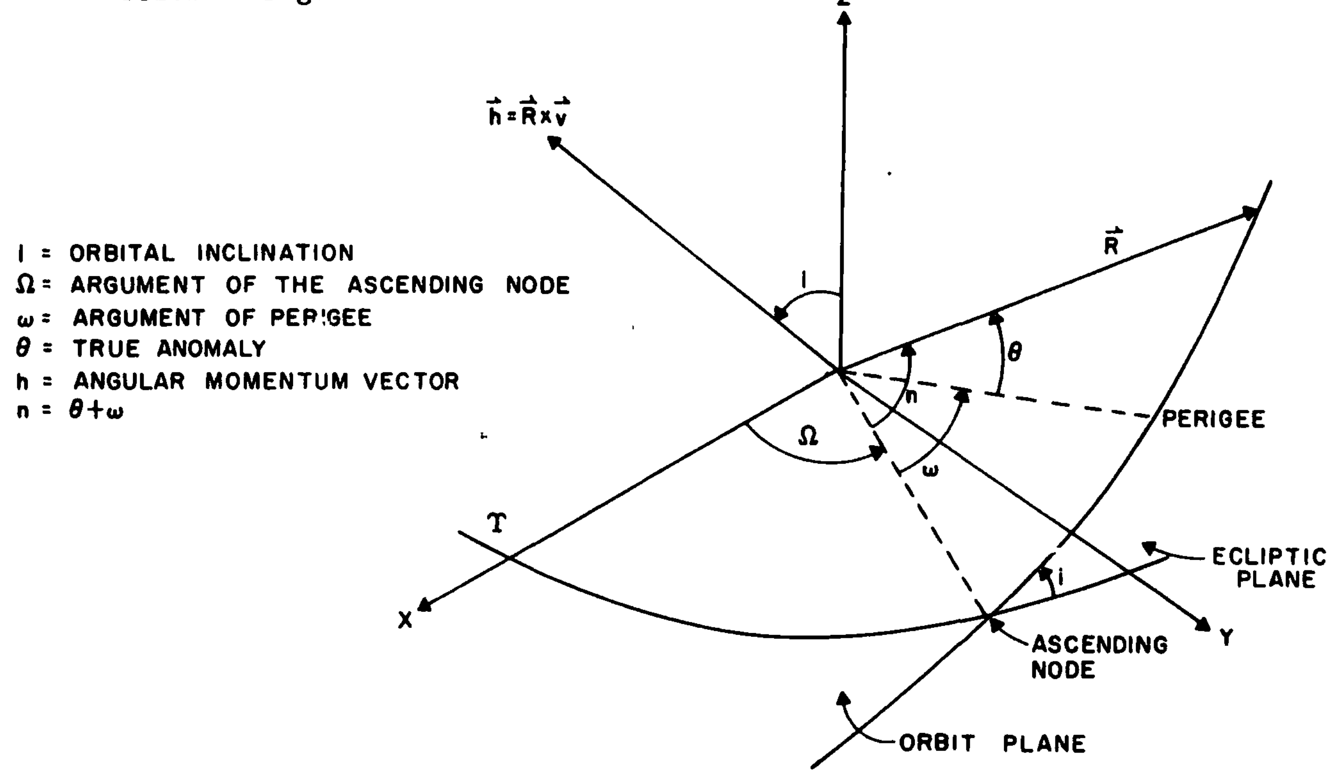


FIG. 1

1. Derivation of partial for eccentricity

$$\epsilon = \sqrt{1 + \frac{h^2 C3}{\mu^2}}$$

$$\nabla \epsilon = \frac{1}{\mu^2 \epsilon} \left\{ C3 \left[v^2 \vec{r} - (\vec{r} \cdot \vec{v}) \vec{v} \right] + h^2 \mu \frac{\vec{r}}{r^3} \right\}$$

$$\text{where } \nabla = \left(\frac{\partial}{\partial x} ; \frac{\partial}{\partial y} ; \frac{\partial}{\partial z} \right)$$

$$C3 = v^2 - \frac{2\mu}{r}$$

$$h^2 = |\vec{r} \times \vec{v}|^2$$

$$D\epsilon = \frac{1}{\mu^2 \epsilon} \left\{ h^2 \vec{v} + C3 \left[r^2 \vec{v} - (\vec{r}\cdot\vec{v}) \vec{r} \right] \right\}$$

where $D = \left(\frac{\partial}{\partial \dot{x}} ; \frac{\partial}{\partial \dot{y}} ; \frac{\partial}{\partial \dot{z}} \right)$

2. Derivation of partials for radius of closest approach

$$r_p = \frac{h^2}{\mu(1+\epsilon)}$$

$$\nabla r_p = \frac{1}{\mu\epsilon} \left\{ v^2 \vec{r} - (\vec{r}\cdot\vec{v}) \vec{v} - \mu\, r_p^2 \frac{\vec{r}}{r^3} \right\}$$

$$Dr_p = \frac{1}{\mu\epsilon} \left\{ \left(r^2 - r_p^2 \right) \vec{v} - (\vec{r}\cdot\vec{v}) \vec{r} \right\}$$

3. Derivation of partials for semi-major axis

$$a = - \frac{\mu}{|C3|}$$

$$\frac{\partial a}{\partial X_k} = 2a^2 \frac{X_k}{r^3} \qquad \text{for } k = 1,2,3$$

$$X_1 = X \qquad X_2 = Y \qquad X_3 = Z$$

$$\frac{\partial a}{\partial \dot{X}_k} = 2a^2 \frac{\dot{X}_k}{\mu} \qquad \dot{X}_1 = \dot{X} \qquad \dot{X}_2 = \dot{Y} \qquad \dot{X}_3 = \dot{Z}$$

4. Derivation of partials for longitude of the ascending node

$$\Omega = \mathrm{Tan}^{-1} \frac{h(1)}{-h(2)}$$

$$\frac{\partial \Omega}{\partial x} = - \sin \Omega \frac{\dot{z}}{h \sin i} \qquad \frac{\partial \Omega}{\partial \dot{x}} = \sin \Omega \frac{z}{h \sin i}$$

$$\frac{\partial \Omega}{\partial y} = \cos \Omega \frac{\dot{z}}{h \sin i} \qquad \frac{\partial \Omega}{\partial \dot{y}} = - \cos \Omega \frac{z}{h \sin i}$$

$$\frac{\partial \Omega}{\partial z} = -\cos i \frac{\dot{z}}{h \sin i} \qquad \frac{\partial \Omega}{\partial \dot{z}} = \cos i \frac{z}{h \sin i}$$

5. Derivation of partials for inclination

$$i = \mathrm{Tan}^{-1} \frac{\sqrt{h(1)^2 + h(2)^2}}{h(3)}$$

$$\frac{\partial i}{\partial X_k} = \frac{\sin i \cos n}{\sin n} \frac{\partial \Omega}{\partial X_k} \qquad \text{for } k = 1,2,\ldots.6$$

$$X_1 = X \qquad X_2 = Y \qquad X_3 = Z$$

$$X_4 = \dot{X} \qquad X_5 = \dot{Y} \qquad X_6 = \dot{Z}$$

6. Derivation of partials for argument of periapsis

$$\omega = n - \theta = \mathrm{Tan}^{-1} \frac{zh}{yh(1) - xh(2)} - \mathrm{Tan}^{-1} \frac{(\vec{r}\cdot\vec{v})\, h}{h^2 - \mu r}$$

$$\frac{\partial \omega}{\partial X_k} = -\frac{1}{\epsilon^2}(\alpha \cos n + \beta \sin n) \frac{\partial n}{\partial X_k} + \frac{1}{h\epsilon^2}\left\{(\alpha \sin n - \beta \cos n)\right\}$$

$$\left\{\frac{\partial h}{\partial X_k} - (\alpha + \cos n) \frac{\cos i}{\sin n} h \frac{\partial \Omega}{\partial X_k}\right\}$$

for k = 1,2,3

$$\frac{\partial \omega}{\partial \dot{x}} = -\frac{1}{\epsilon^2}(\alpha \cos n + \beta \sin n) \frac{\partial n}{\partial \dot{x}} + \frac{1}{h\epsilon^2}(\alpha \sin n - \beta \cos n)\left\{\frac{\partial h}{\partial \dot{x}}\right.$$

$$\left. -(\alpha + \cos n) \frac{\cos i}{\sin n} h \frac{\partial \Omega}{\partial \dot{x}}\right\} - \frac{h}{\epsilon^2 \mu} \alpha \cos \Omega$$

$$\frac{\partial \omega}{\partial \dot{y}} = -\frac{1}{\epsilon^2}(\alpha \cos n + \beta \sin n)\frac{\partial n}{\partial \dot{y}} + \frac{1}{h\epsilon^2}(\alpha \sin n - \beta \cos n)\left\{\frac{\partial h}{\partial \dot{y}} - (\alpha + \cos n)\frac{\cos i}{\sin n}\, h\, \frac{\partial \Omega}{\partial \dot{y}}\right\} - \frac{1}{\mu\epsilon^2}\alpha \sin \Omega$$

$$\frac{\partial \omega}{\partial \dot{z}} = -\frac{1}{\epsilon^2}(\alpha \cos n + \beta \sin n)\frac{\partial n}{\partial \dot{z}} + \frac{1}{h\epsilon^2}(\alpha \sin n - \beta \cos n)\left\{\frac{\partial h}{\partial \dot{z}} - (\alpha + \cos n)\frac{\cos i}{\sin n}\, h\, \frac{\partial \Omega}{\partial \dot{z}}\right\} - \frac{h}{\mu\epsilon^2}\ \frac{\beta}{\sin i}$$

where

$$\alpha = \epsilon \cos \omega \qquad\qquad \beta = \epsilon \sin \omega$$

$$\frac{\partial h}{\partial X_k} = \frac{h \sin^2 i \cos n}{\cos i \sin n}\ \frac{\partial \Omega}{\partial X_k} \qquad k = 1,2,\ldots 6$$

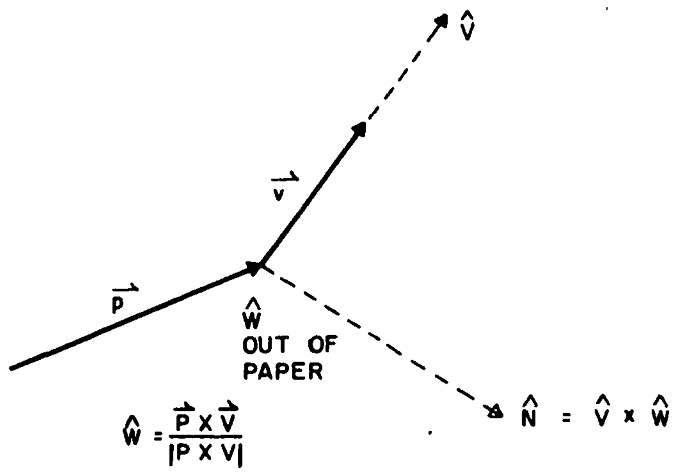
$$\frac{\partial n}{\partial x} = -\cos i\,\frac{\partial \Omega}{\partial x} - \frac{1}{r}\sin n \cos \Omega \qquad\qquad \frac{\partial n}{\partial \dot{x}} = -\cos i\,\frac{\partial \Omega}{\partial \dot{x}}$$

$$\frac{\partial n}{\partial y} = -\cos i\,\frac{\partial \Omega}{\partial y} - \frac{1}{r}\sin n \sin \Omega \qquad\qquad \frac{\partial n}{\partial \dot{y}} = -\cos i\,\frac{\partial \Omega}{\partial \dot{y}}$$

$$\frac{\partial n}{\partial z} = -\cos i\,\frac{\partial \Omega}{\partial z} + \frac{1}{r}\ \frac{\cos n}{\sin i} \qquad\qquad \frac{\partial n}{\partial \dot{z}} = -\cos i\,\frac{\partial \Omega}{\partial \dot{z}}$$

B. <u>Derivation of Transformation from Cartesian x, y, z to Cartesian N, V, W coordinates</u>

The coordinates N, V, and W represent unit vectors shown in Fig. 2 following.



(PAPER IS ORBIT PLANE)

$\vec{p}$ AND $\vec{v}$ ARE POSITION AND VELOCITY VECTORS

FIG. 2

$\hat{W}$ is in the direction of the angular momentum vector, $\hat{V}$ is along the velocity vector, and $\hat{N}$ forms an orthogonal coordinate system in the sense of $\hat{N} = \hat{V} \times \hat{W}$. The three unit vectors may be written as follows:

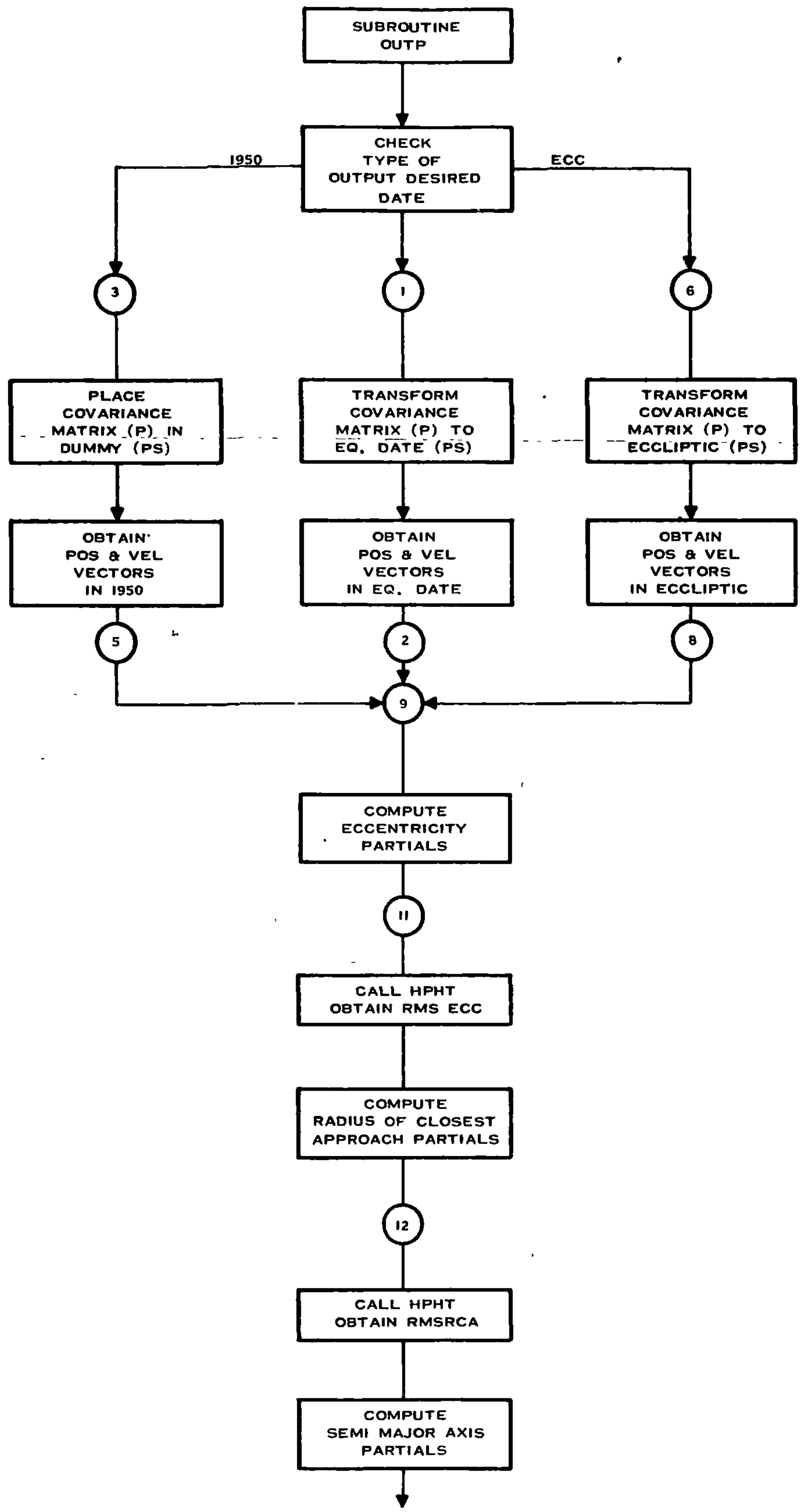
$$\hat{W} = \frac{\vec{R} \times \vec{v}}{|\vec{R} \times \vec{v}|}$$

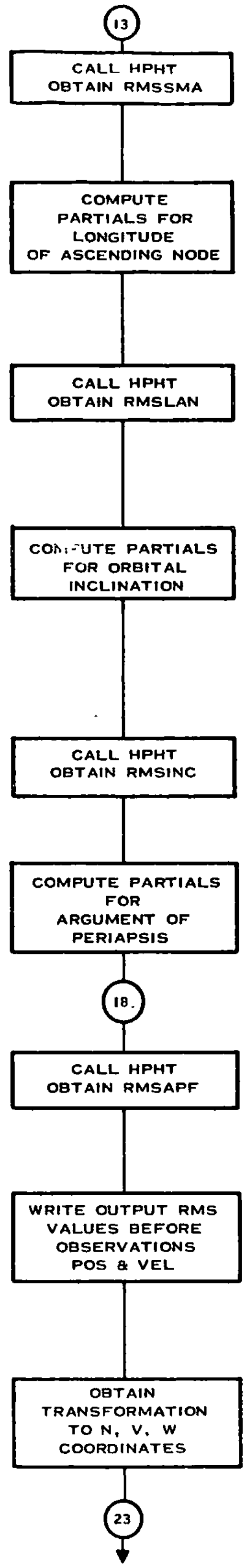
$$\hat{V} = \frac{\vec{v}}{|\vec{v}|}$$

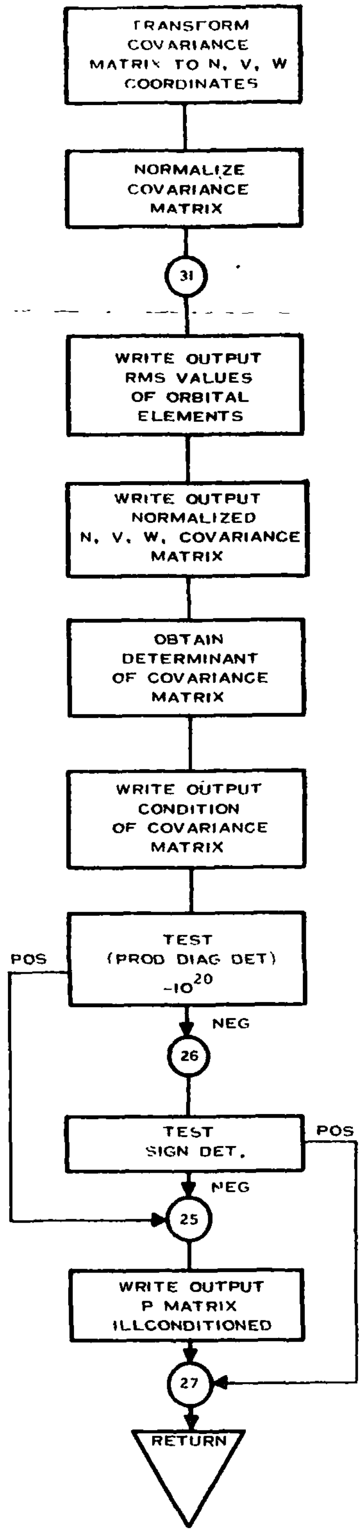
$$\hat{N} = \frac{\vec{v}}{|\vec{v}|} \times \frac{\vec{R} \times \vec{v}}{|\vec{R} \times \vec{v}|}$$

Writing the transposed vectors in matrix form yields the desired transformation from $\hat{i}$, $\hat{j}$, $\hat{k}$ to $\hat{N}$, $\hat{V}$, $\hat{W}$

$$\underset{3\text{x}1}{\begin{pmatrix} N \\ V \\ W \end{pmatrix}} = \underset{3\text{x}3}{\begin{pmatrix} \hat{N}^T \\ \hat{V}^T \\ \hat{W}^T \end{pmatrix}} \underset{3\text{x}1}{\begin{pmatrix} x \\ y \\ z \end{pmatrix}}$$


SUBROUTINE OUTP
CHECK TYPE OF OUTPUT DESIRED DATE
1950
ECC
3
1
6
PLACE COVARIANCE MATRIX (P) IN DUMMY (PS)
TRANSFORM COVARIANCE MATRIX (P) TO EQ. DATE (PS)
TRANSFORM COVARIANCE MATRIX (P) TO ECCLIPTIC (PS)
OBTAIN POS & VEL VECTORS IN 1950
OBTAIN POS & VEL VECTORS IN EQ. DATE
OBTAIN POS & VEL VECTORS IN ECCLIPTIC
5
2
8
9
COMPUTE ECCENTRICITY PARTIALS
11
CALL HPHT OBTAIN RMS ECC
COMPUTE RADIUS OF CLOSEST APPROACH PARTIALS
12
CALL HPHT OBTAIN RMSRCA
COMPUTE SEMI MAJOR AXIS PARTIALS


13
CALL HPHT OBTAIN RMSSMA
COMPUTE PARTIALS FOR LONGITUDE OF ASCENDING NODE
CALL HPHT OBTAIN RMSLAN
COMPUTE PARTIALS FOR ORBITAL INCLINATION
CALL HPHT OBTAIN RMSINC
COMPUTE PARTIALS FOR ARGUMENT OF PERIAPSIS
18
CALL HPHT OBTAIN RMSAPF
WRITE OUTPUT RMS VALUES BEFORE OBSERVATIONS POS & VEL
OBTAIN TRANSFORMATION TO N, V, W COORDINATES
23


TRANSFORM COVARIANCE MATRIX TO N, V, W COORDINATES
NORMALIZE COVARIANCE MATRIX
31
WRITE OUTPUT RMS VALUES OF ORBITAL ELEMENTS
WRITE OUTPUT NORMALIZED N, V, W, COVARIANCE MATRIX
OBTAIN DETERMINANT OF COVARIANCE MATRIX
WRITE OUTPUT CONDITION OF COVARIANCE MATRIX
TEST (PROD DIAG DET) -10^20
POS
NEG
26
TEST SIGN DET.
POS
NEG
25
WRITE OUTPUT P MATRIX ILLCONDITIONED
27
RETURN

```
*       LABEL                                                                  OUTP
CEC2016  SUBROUTINE OUTP                                                       OUTP
        SUBROUTINE OUTP                                                        OUTP0000
        COMMON T,S,C,IC                                                        OUTP0010
        DIMENSION T(1360),S(1000),C(1000),IC(1),P(6,6),AN(3,3),ECL(3,3)        OUTP0020
       1,ECX(3),ECV(3),XED(3),VED(3),XP(3),VXP(3),PS(6,6),X(3),VX(3)           OUTP0030
       2,DUM(3,3),  H (6),DUMC(3,3),DELH2(3),DELC3(3),DELECC(3),DELVH2(3)      OUTP0040
       3,DELVEC(3),DELRP(3),DELVRP(3),DELSMA(3),DELVSA(3),Q(6)                 OUTP0050
        DIMENSION Q1(6),RCV(3),DAG(6)                                          OUTP0060
        EQUIVALENCE (T,TDUM),(S,SDUM),(C,CDUM),(IC,ICDUM)                      OUTP0070
       1,(C(652),P  ),(C(138),AN),(C(147),ECL),(C(156),ECX),(C(159),ECV)       OUTP0080
       2,(C(180),SMA),(C(181),ECC),(C(182),RCA),(C(183),OINC),(C(184),OMG)OUTP0090
       3,(C(185),APF),(C(186),XED),(C(189),VED),(C(192),RCV),(C(195),RCV2)OUTP0100
       4,(C(196),R2),(C(197),V2) ,(C(198),RDV),(C(199),C3),(C(15),XP)          OUTP0110
       5,(C(18),VXP),(IC(7),KOUT),(C(55) ,U),(C(162),BET),(C(163),THT)         OUTP0120
       6,(S(73),RTD)                                                           OUTP0130
        CALL SETN(NIN,NUTS)                                                    OUTP0138
C       KOUT=TYPE OUTPUT DESIRED  0=EQ OF DATE 1= EQ OF 1950 2=ECCLIPTIC       OUTP0140
        IF(KOUT-1) 1,3,6                                                       OUTP0150
    1   CONTINUE                                                               OUTP0160
        CALL PTRAN(P,AN,PS,2)                                                  OUTP0170
        DO 2 I=1,3                                                             OUTP0180
        X(I)=XED(I)                                                            OUTP0190
    2   VX(I)=VED(I)                                                           OUTP0200
        GO TO 9                                                                OUTP0210
    3   CONTINUE                                                               OUTP0220
        DO 4 I=1,6                                                             OUTP0230
        DO 4 J=1,6                                                             OUTP0240
    4   PS(I,J)=P(I,J)                                                         OUTP0250
        DO 5 I=1,3                                                             OUTP0260
        X(I)=XP(I)                                                             OUTP0270
    5   VX(I)=VXP(I)                                                           OUTP0280
        GO  TO 9                                                               OUTP0290
    6   CONTINUE                                                               OUTP0300
        DO 7 I=1,3                                                             OUTP0310
        DO 7 J=1,3                                                             OUTP0320
        DUM(I,J)=0.                                                            OUTP0330
        DO 7 K=1,3                                                             OUTP0340
    7   DUM(I,J)=DUM(I,J) + ECL(I,K)*AN(K,J)                                   OUTP0350
        CALL PTRAN( P,DUM,PS,2)                                                OUTP0360
        DO 8 I=1,3                                                             OUTP0370
        X(I)=ECX(I)                                                            OUTP0380
    8   VX(I)=ECV(I)                                                           OUTP0390
    9   CONTINUE                                                               OUTP0400
        R=SQRTF(R2)                                                            OUTP0410
        CALL CROSS(VX,RCV,DELH2)                                               OUTP0420
        CALL CROSS(RCV,X,DELVH2)                                               OUTP0430
        DO 10 I=1,3                                                            OUTP0440
   10   DELC3(I)=(U/R2/R)*X(I)                                                 OUTP0450
C       FOLLOWING IS COMPUTATION  OF  ECCENTRICITY PARTIALS                    OUTP0460
        CONST= 1./(ECC*U**2)                                                   OUTP0470
```

```
      DO 11 I=1,3                                                      OUTP048
      II=I+3                                                           OUTP049
      DELECC(I)=CONST*(C3*DELH2(I)+RCV2*DELC3(I))                      OUTP050
      DELVEC(I)=CONST*(C3*DELVH2(I)+RCV2*VX(I))                        OUTP051
      H(I)=DELECC(I)                                                   OUTP052
      H(II)=DELVEC(I)                                                  OUTP053
   11 CONTINUE                                                         OUTP054
      CALL HPHT(H,PS,RMSECC)                                           OUTP055
C     FOLLOWING FOR PARTIALS OF RADIUS CLOSEST APPROACH =RCA           OUTP056
      CON=1.+ECC                                                       OUTP057
      CONST=U*CON                                                      OUTP058
      CONS =RCV2/(CONST*CON)                                           OUTP059
      CON=2./CONST                                                     OUTP060
      DO 12 I=1,3                                                      OUTP061
      II= I+3                                                          OUTP062
      DELRP(I)=CON*DELH2(I)-CONS*DELECC(I)                             OUTP063
      DELVRP(I)=CON*DELVH2(I)-CONS*DELVEC(I)                           OUTP064
      H(I)=DELRP(I)                                                    OUTP065
      H(II)=DELVRP(I)                                                  OUTP066
   12 CONTINUE                                                         OUTP067
      CALL HPHT(H,PS,RMSRCA)                                           OUTP068
C     FOLLOWING FOR PARTIALS OF SEMI MAJOR AXIS =SMA                   OUTP069
      CONST= 1./( ECC-1.)                                              OUTP070
      CON= RCA*CONST**2                                                OUTP071
      DO 13 I=1,3                                                      OUTP072
      II=I+3                                                           OUTP073
      DELSMA(I)=CONST*DELRP(I)-CON*DELECC(I)                           OUTP074
      DELVSA(I)=CONST*DELVRP(I)-CON*DELVEC(I)                          OUTP075
      H(I)=DELSMA(I)                                                   OUTP076
      H(II)=DELVSA(I)                                                  OUTP077
   13 CONTINUE                                                         OUTP078
      CALL HPHT(H,PS,RMSSMA)                                           OUTP084
      CI=COSF(OINC)                                                    OUTP085
      SI=SINF(OINC)                                                    OUTP086
      CW=COSF(OMG)                                                     OUTP087
      SW=SINF(OMG)                                                     OUTP088
C     FOLLOWING FOR LINE OF MODES =LAN  PARTIALS                       OUTP089
      CON =SQRTF(RCV2)                                                 OUTP090
      CONST=1./(CON*SI)                                                OUTP091
      CONS =VX(3)*CONST                                                OUTP092
      G(1) =-SW*CONS                                                   OUTP093
      G(2) =CW *CONS                                                   OUTP094
      G(3) =-CI*CONS/SI                                                OUTP095
      CONS =X(3)*CONST                                                 OUTP096
      G(4) =SW*CONS                                                    OUTP097
      G(5) =-CW*CONS                                                   OUTP098
      G(6) = CI*CONS/SI                                                OUTP099
      CALL HPHT(G,PS,RMSLAN)                                           OUTP100
C     THE FOLLOWING FOR INCLINATION                                    OUTP101
      CONS=1./CON                                                      OUTP102
      H(1)=CONS*(-VX(2)*SI + VX(3)*CW*CI)                              OUTP103
```

```
      H(2)=CONS*(VX(1)*SI   + VX(3)*SW*CI)                                  OUTP1040
      H(3)=-CONS*CI*(VX(1)*CW+VX(2)*SW)                                     OUTP1050
      H(4)=CONS*(X(2)*SI-X(3)*CW*CI)                                        OUTP1060
      H(5)=-CONS*(X(1)*SI+X(3)*SW*CI)                                       OUTP1070
      H(6)= CONS*CI*(X(1)*CW + X(2)*SW)                                     OUTP1080
      CALL HPHT(H,PS,RMSINC)                                                OUTP1090
C   THE FOLLOWING ARE PARTIALS OF ARG. OF PERIAPSIS  = APF                  OUTP1100
      SN =SINF(BET)                                                         OUTP1110
      CN = COSF(BET)                                                        OUTP1120
      DO 17 I=1,6                                                           OUTP1130
      Q1(I) = -CI*Q(I)                                                      OUTP1140
   17 CONTINUE                                                              OUTP1150
      Q1(1) = Q1(1)-SN*CW/R                                                 OUTP1160
      Q1(2) = Q1(2)-SN*SW/R                                                 OUTP1170
      Q1(3) = Q1(3)+CN/(R*SI)                                               OUTP1180
      CP = COSF(APF)                                                        OUTP1190
      SP = SINF(APF)                                                        OUTP1200
      RAGHI = 1./SQRTF(RCV2)                                                OUTP1210
      CON = ECC*CP                                                          OUTP1220
      CONST = ECC*SP                                                        OUTP1230
      CONS = -(CON*CN+CONST*SN)/(ECC**2)                                    OUTP1240
      CONS1 = (CON*SN-CONST*CN)*RAGHI/(ECC**2)                              OUTP1250
      CONS2 = CON+CN                                                        OUTP1260
      CONS2 = CONS2*RAGHI*CI/SN                                             OUTP1270
      DO 18 I=1,3                                                           OUTP1280
      II = I+3                                                              OUTP1290
      H(I) = CONS*Q1(I)+CONS1*(RAGHI*DELH2(I)-CONS2*Q(I))                   OUTP1300
      H(II) = CONS*Q1(II)+CONS1*(RAGHI*DELVH2(I)-CONS2*Q(II))               OUTP1310
   18 CONTINUE                                                              OUTP1320
      CONS = -1./((RAGHI*U)*(ECC**2))                                       OUTP1330
      H(4)= H(4)+CONS*CW*CON                                                OUTP1340
      H(5)= H(5)+CONS*SW*CON                                                OUTP1350
      H(6) = H(6)+CONS*CONST/SI                                             OUTP1360
      CALL HPHT(H,PS,RMSAPF)                                                OUTP1370
      WRITE OUTPUT TAPE NUTS,700                                            OUTP1380
  700 FORMAT( 44X,31H RMS VALUES BEFORE OBSERVATIONS)                       OUTP1390
      DO 21 I=1,6                                                           OUTP1400
   21 H(I)    =SQRTF(PS(I,I))                                               OUTP1410
      WRITE OUTPUT TAPE NUTS,701,H                                          OUTP1420
  701 FORMAT(4H   XE15.8,5H    YE15.8,5H    ZE15.8,5H   DXE15.8,            OUTP1430
     15H   DYE15.8,5H   DZE15.8)                                            OUTP1440
      R50=FNORM(VXP)                                                        OUTP1580
      CALL CROSS(XP,VXP,DUM(1,3))                                           OUTP1590
      R51=FNORM(DUM(1,3))                                                   OUTP1600
      DO 22 I=1,3                                                           OUTP1610
      DUM(I,2)=VXP(I)/R50                                                   OUTP1620
   22 DUM(I,3)= DUM(I,3)/R51                                                OUTP1630
      CALL CROSS(DUM(1,2),DUM(1,3),DUM(1,1))                                OUTP1640
      DO 23 I=1,3                                                           OUTP1650
      DO 23 J=1,3                                                           OUTP1660
   23 DUMC(I,J)=DUM(J,I)                                                    OUTP1670
```

```
      CALL PTRAN(P,DUMC,PS,2)                                          OUTP168
      DO 30 I=1,6                                                      OUTP169
      DAG(I)=SQRTF(PS(I,I))                                            OUTP170
 30   CONTINUE                                                         OUTP171
      DO 31 I=1,6                                                      OUTP172
      DO 32 J=I,6                                                      OUTP173
      PS(I,J)=PS(I,J)/DAG(I)/DAG(J)                                    OUTP174
 32   CONTINUE                                                         OUTP175
      PS(I,I)=1.                                                       OUTP176
 31   CONTINUE                                                         OUTP177
      RMSINC = RMSINC*RTD                                              OUTP178
      RMSAPF = RMSAPF*RTD                                              OUTP179
      RMSLAN = RMSLAN*RTD                                              OUTP180
     0WRITE OUTPUT TAPE NUTS,702,RMSSMA,RMSECC,RMSINC,RMSLAN,RMSAPF,   OUTP181
     1RMSRCA                                                           OUTP181
 702  FORMAT(4H SMAE15.8,5H  ECCE15.8,5H  INCE15.8,5H  LANE15.8,       OUTP182
     15H  APFE15.8,5H  RCAE15.8)                                       OUTP183
      WRITE OUTPUT TAPE NUTS,703,DAG                                   OUTP184
 703  FORMAT(50X,22H RMS N,V,W COORDINATES,/21X,17H RMS POSITION, KM,  OUTP185
     142X,20H RMS VELOCITY KM/SEC,/7X,7H NORMAL, 12X,9H VELOCITY,11X,  OUTP186
     29H MOMENTUM,12X,7H NORVEL,14X,7H VELVEL,11X,9H MOMENVEL,/4H    E15OUTP187
     3.8,(5E20.8))                                                     OUTP188
      WRITE OUTPUT TAPE NUTS,704,((PS(I,J),J=1,6),I=1,6)               OUTP189
 704 FORMAT(30X,58H NORMALIZED STATE COVARIANCE MATRIX IN  N,V,W  COORDOUTP190
     1INATES,/4H    E15.8,(5E20.8)/19X,(5E20.8)/39X,(4E20,8)/59X,      OUTP191
     2(3E20.8)/79X,(2E20.8)/99X,E20.8)                                 OUTP192
      CON = 1.                                                         OUTP192
      CALL INV3(PS,6,DETER)                                            OUTP192
      CONST = CON/DETER                                                OUTP192
      WRITE OUTPUT TAPE NUTS,705,DETER,CON,CONST                       OUTP193
 705 FORMAT(37H CONDITION OF STATE COVARIANCE MATRIX,/                 OUTP194
     124H DETERMINANT OF MATRIX =,E15.8,3X,                            OUTP195
     222H PRODUCT OF DIAGONAL =,E15.8,1X,                              OUTP196
     324H RATIO PROD.DIAG,/DET. =,E15.8)                               OUTP197
      IF (ABSF(CONST)-1.E20) 26,25,25                                  OUTP198
  26 IF (DETER) 25,25,27                                               OUTP199
      WRITE OUTPUT TAPE NUTS,706                                       OUTP200
 706 FORMAT(33H ***P MATRIX IS ILLCONDITIONED***)                      OUTP201
  27 CONTINUE                                                          OUTP202
      RETURN                                                           OUTP204
      END                                                              OUTP
```

Page Intentionally Left Blank

Subroutine: PTRAN

Purpose: To compute the transformation of the state covariance matrix from one inertial coordinate system to another. The transformation which is obtained with KK = 2 is the following:

$$\underset{6\times6}{PS} = \underset{6\times6}{\begin{pmatrix} B & 0 \\ 0 & B \end{pmatrix}} \underset{6\times6}{\begin{pmatrix} P \end{pmatrix}} \underset{6\times6}{\begin{pmatrix} B^T & 0 \\ 0 & B^T \end{pmatrix}}$$

If KK is set 1 only, the diagonal 3x3's are computed.

Calling Sequence:

CALL PTRAN (P, B, PS, KK)

Input and **Output**

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	P	(6,6)			Covariance Matrix
I	B	(3,3)			Transformation Matrix
O	PS	(6,6)			Transformed Matrix
I	KK	1			Logic Key

Common storages used or required: None

Subroutines required: None

Functions required: None

Approximate number of storages required:

```
*       LABEL
*       SYMBOL TABLE
        SUBROUTINE PTRAN( P,B,PS,KK)
        DIMENSION P(6,6),B(3,3),PS(6,6),DUM(6,6)
C       SUBROUTINE CALCULATES THE 6X6 MATRIX PS  FROM
C       P IS SYMMETRIC PS=( B   0 )(   P )( BT   0 )    T=TRANSPOSE
C           ( 0   B )(     )( 0    BT)    K=1 OBTAINS ONLY DIAG 3X3
        DO 1  I=1,6
        DO 1  J=1,6
        PS(I,J)=0.
    1   DUM(I,J)=0.
        DO 2 I=1,3
        I3=I+3
        DO 2 J=1,3
        J3=J+3
        DO 2 K=1,3
        K3 = K+3
        DUM(I,J)=DUM(I,J)+  B(I,K)*P(K,J)
        DUM(I3,J3)=DUM(I3,J3)+  B(I,K)*P(K3,J3)
        GO TO(2,3), KK
    3   DUM(I,J3) =DUM(I,J3) +  B(I,K)*P(K,J3)
    2   CONTINUE
        DO 5 I=1,3
        I3=I+3
        DO 5 J=1,3
        J3=J+3
        DO 4 K=1,3
        K3=K+3
        PS(I,J)=PS(I,J) + DUM(I,K)*B(J,K)
        PS(I3,J3)=PS(I3,J3) +DUM(I3,K3)*B(J,K)
        GO TO(4,6), KK
    6   PS(I,J3) =PS(I,J3)  +DUM(I ,K3)*B(J,K)
    4   CONTINUE
        GO TO(5,7),KK
    7   PS(J3,I)=PS(I,J3)
    5   CONTINUE
    8    CONTINUE
        RETURN
        END
```

Subroutine: RETRO

Purpose: To take the P and PAR covariance matricies through a retro maneuver into a circular orbit.

Calling Sequence:

CALL RETRO

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I					Quantities obtained from common
O					Quantities returned to common

Common storages used or required: T, S, C, IC

Subroutines required: MATRX, SETN

Functions required: DOT, FNORM, SQRT

Approximate number of storages required: 779 DEC

Derivation of Guidance Law Through Retro Maneuver

The vehicle is assumed to be at periapsis at the time the subroutine is called. The maneuver is then carried out such that the vehicle is in circular orbit after retro fire.

The required velocity at periapsis for a circular orbit is:

$$\upsilon_p = \sqrt{\frac{\mu}{R_p}}$$

The required incremental velocity correction along the nominal trajectory is the following

$$\Delta \vec{V} = \left(\frac{\upsilon_p}{\upsilon_n} - 1\right) \vec{\upsilon}_n \qquad \upsilon_n = |\vec{\upsilon}_n| \tag{1}$$

where υ_n is the nominal periapsis velocity vector for the incoming trajectory.

The guidance law which governs the deviations from the nominal is obtained in the following manner. The deviation from nominal periapsis velocity (assuming linear theory) may be written as:

$$\underset{3x1}{\delta \vec{\upsilon}_p} = \underset{3x1}{\dot{\underline{X}}} + \underset{(3x3)}{A}\ \underset{(3x1)}{\underline{X}} \tag{2}$$

where

$$A = \begin{pmatrix} \frac{\partial \upsilon_p}{\partial X_1} & 0 & 0 \\ 0 & \frac{\partial \upsilon_p}{\partial X_2} & 0 \\ 0 & 0 & \frac{\partial \upsilon_p}{\partial X_3} \end{pmatrix}$$

and $\underline{X}$ and $\dot{\underline{X}}$ are the vehicle deviation state.

The partial derivatives in matrix A are the following:

$$\frac{\partial \upsilon_p}{\partial X_i} = - \frac{\mu X_i}{2\upsilon_p R^3} \qquad i = 1,2,3$$

The estimate of periapsis velocity deviation from equation (2) is

$$\delta \hat{\dot{\upsilon}}_p = \hat{\dot{\underline{X}}} + A \hat{\underline{X}} \tag{3}$$

and the guidance law is obtained as follows:

$$\delta \hat{\dot{\upsilon}}_p + \dot{\underline{X}}_g = 0 = \hat{\dot{\underline{X}}} + A \hat{\underline{X}} + \dot{\underline{X}}_g$$

or

$$\underset{3x1}{\dot{\underline{X}}_g} = - \underset{3x6}{(A \;\; I)} \underset{6x1}{\begin{pmatrix} \hat{\underline{X}} \\ \hat{\dot{\underline{X}}} \end{pmatrix}} \tag{4}$$

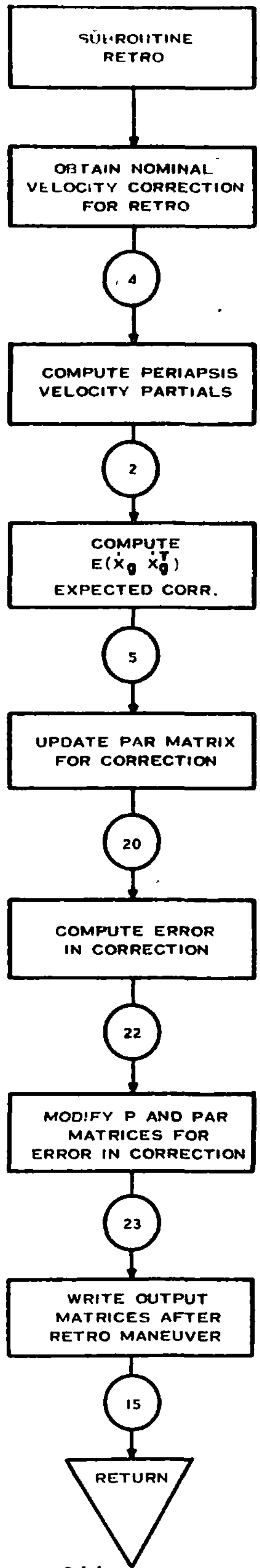
The total guidance correction is the sum of equations (1) and (4).

$$\dot{\underline{X}}_{gt} = - (A \;\; I) \begin{pmatrix} \hat{\underline{X}} \\ \hat{\dot{X}} \end{pmatrix} + \overrightarrow{\Delta V} \tag{5}$$

The guidance law expressed by equation (4) corresponds to equation (6) in the guidance correction derivation in subroutine GUID. The P and PAR covariance matricies are then updated as shown by the derivation in the GUID subroutine writeup. The additional term, $\Delta \vec{V}$, present in the total guidance correction required is treated as being independent of the deviation state. Therefore, the E $(\dot{\underline{X}}_{gt} \; \dot{X}^T_{gt})$ may be written as:

$$E\,(\dot{\underline{X}}_{gt} \; \dot{\underline{X}}^T_{gt}) = E\,(\dot{X}_g \; \dot{X}^T_g) + \overrightarrow{\Delta V} \; \overrightarrow{\Delta V}^T$$

$$\text{since } E(\dot{X}_g \; \overrightarrow{\Delta V}^T) = 0$$


SUBROUTINE RETRO
OBTAIN NOMINAL VELOCITY CORRECTION FOR RETRO
4
COMPUTE PERIAPSIS VELOCITY PARTIALS
2
COMPUTE $E(\dot{X}_g \dot{X}_g^T)$ EXPECTED CORR.
5
UPDATE PAR MATRIX FOR CORRECTION
20
COMPUTE ERROR IN CORRECTION
22
MODIFY P AND PAR MATRICES FOR ERROR IN CORRECTION
23
WRITE OUTPUT MATRICES AFTER RETRO MANEUVER
15
RETURN

```
*       LABEL                                                            RETR
*       SYMBOL TABLE                                                     RETR
CEC2030  SUBROUTINE RETRO                                                RETR
        SUBROUTINE RETRO                                                 RETR000
        COMMON T,S,C,IC                                                  RETR001
        DIMENSION T(1360),S(1000),C(1000),IC(1)                          RETR002
       1,TPAR(3,6),PAR(6,6),P(6,6),DUP(3,3)                              RETR003
       2,TGUID(6),DUM(3,3),DUN(3,3),DUMM(6,6)                            RETR004
        DIMENSION XP(3),VXP(3),CV(3)                                     RETR005
        EQUIVALENCE (T,TDUM),(S,SDUM),(C,CDUM)                           RETR006
       1,(IC,ICDUM),(C(568),TPAR),(C(460),PAR)                           RETR007
       2,(C(652),P),(IC(218),N),(C(648),TGUIDE)                          RETR008
       3,(IC(219),IGD),(IC(217),IGDTP),(IC(214),NOUT)                    RETR009
       4,(C(30),TSEC),(S(476),TGUID),(S(483),RATIO)                      RETR010
       5,(S(473),QQPT),(S(471),QQSO),(S(484),DMONIT)                     RETR011
        EQUIVALENCE (C(15),XP),(C(18),VXP),(C(55),U)                     RETR012
        CALL SETN(NIN,NUTS)                                              RETR013
        WRITE OUTPUT TAPE NUTS,701                                       RETR014
  701   FORMAT(94H THE DOLLOWING DATA ARE CONDITIONS AFTER RETRO INTO CIRCRETR015
       1ULAR ORBIT ASSUMING VEHICLE AT PERIGEE)                          RETR016
        R2=DOT(XP,XP)                                                    RETR017
        R=FNORM(XP)                                                      RETR018
        VM=FNORM(VXP)                                                    RETR019
        VP=SQRTF(U/R)                                                    RETR020
        DELV=VP/VM                                                       RETR021
        DO 1 I=1,3                                                       RETR022
    1   CV(I)=DELV*VXP(I)                                                RETR023
        WRITE OUTPUT TAPE NUTS,700,CV                                    RETR024
  700   FORMAT(35H CIRCULAR VELOCITY COMPONENTS X,Y,Z/(3E17.8))          RETR025
        DO 4 I=1,3                                                       RETR026
    4   CV(I)=(CV(I)-VXP(I))**2                                          RETR027
        CONS=-U/(2.*VP*R2*R)                                             RETR028
        DO 2 I=1,3                                                       RETR029
        II=I+3                                                           RETR030
        DO 3 J=1,3                                                       RETR031
        JJ=J+3                                                           RETR032
        TPAR(I,J)=0.                                                     RETR033
    3   TPAR(I,JJ)=0.                                                    RETR034
        TPAR(I,I)=CONS*XP(I)                                             RETR035
    2   TPAR(I,II)=-1.                                                   RETR036
        CALL MATRX(TPAR,P,DUM,3,6,6,1)                                   RETR037
        CALL MATRX(TPAR,PAR,DUN,3,6,6,1)                                 RETR038
        DO 5 I=1,3                                                       RETR039
        DO 6 J=1,3                                                       RETR040
    6   DUM(I,J)=DUN(I,J)-DUM(I,J)                                       RETR041
    5   DUM(I,I)=DUM(I,I)+CV(I)                                          RETR042
        RMSV2=DUM(1,1)+DUM(2,2)+DUM(3,3)                                 RETR043
        RMSV=SQRTF(RMSV2)                                                RETR044
C       FOLLOWING IS UPDATING OF PAR MATRIX FOR GUIDANCE CORRECTION      RETR045
        DO 17 I=1,6                                                      RETR046
        DO 17 J=I,6                                                      RETR047
```

```
      PAR(I,J)=PAR(I,J)-P(I,J)
      PAR(J,I)=PAR(I,J)
 17   CONTINUE
      DO 19 I=1,3
      II=I+3
      DO 18 J=1,3
      JJ=J+3
      DUMM(II,J)=TPAR(I,J)
      DUMM(I,JJ)=0.
      DUMM(II,JJ)=0.
 18   DUMM(I,J)=0.
 19   DUMM(I,I)=1.
      CALL MATRX(DUMM,PAR,PAR,6,6,6,1)
      DO 20I=1,6
      DO 20 J=I,6
      PAR(I,J)=PAR(I,J)+P(I,J)
      PAR(J,I)=PAR(I,J)
 20   CONTINUE
      CRUD=QQPT*RMSV2
      QQ=QQSO-QQPT
      DO 22 I=1,3
      DO 21 J=I,3
      DUN(I,J)=QQ*DUM(I,J)
      DUN(J,I)=DUN(I,J)
 21   CONTINUE
      DUN(I,I)=DUN(I,I)+CRUD
 22   CONTINUE
      RMSERR=SQRTF(DUN(1,1)+DUN(2,2)+DUN(3,3))
      DO 23 I=1,3
      II=I+3
      DO 23 J=I,3
      JJ=J+3
      PAR(II,JJ)=PAR(II,JJ)+DUN(I,J)
      PAR(JJ,II)=PAR(II,JJ)
      P(II,JJ)=P(II,JJ)+DMONIT*DUN(I,J)
      P(JJ,II)=P(II,JJ)
 23   CONTINUE
      WRITE OUTPUT TAPE NUTS,702,RMSV,RMSERR
 702  FORMAT(24H RETRO GUID DATA FOLLOWS/
     115H   RMS VEL REQ=,E15.8,10X,18H   RMSERR IN CORR=,E15.8)
      WRITE OUTPUT TAPE NUTS,703,P
 703  FORMAT(9H P MATRIX/(6E17.8))
      WRITE OUTPUT TAPE NUTS,704,PAR
 704  FORMAT(11H PAR MATRIX/(6E17.8))
 15   RETURN
      END
```

Subroutine: ROTATE

Purpose: To compute a 3 x 3 rotation matrix for a transformation of coordinates about axis i (i = 1,2,3) through an angle $\pm\theta$.

For example: rotation about axis 1, matrix = $\begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{pmatrix}$

Calling Sequence:

CALL ROTATE (NCOORD, ANG, RMTRIX, IROT)

Input and **Output**

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	NCOORD	1	i, i = 1,2,3		Axis of rotation
I	ANG	1	θ	radians	Angle for rotation
O	RMTRIX	3,3			Resulting rotation matrix
I	IROT	1			If IROT < 0, use $-\theta$.

Common storages used or required: None

Subroutines required: None

Functions required: None

Approximate number of storages required:

```
*      LABEL
*      SYMBOL TABLE
       SUBROUTINE ROTATE(NCOORD,ANG,RMTRIX,IROT)
       DIMENSION RMTRIX(3,3)
       SANG = SINF(ANG)
       CANG = COSF(ANG)
       IF (IROT) 1,2,2
     1 SANG = -SANG
     2 DO 3 I=1,3
       DO 3 J=1,3
     3 RMTRIX(I,J) = 0.
       GO TO (4,5,6), NCOORD
     4 I = 2
       J = 3
       GO TO 7
     5 I = 1
       J = 3
       GO TO 7
     6 I = 1
       J = 2
     7 RMTRIX(NCOORD,NCOORD) = 1.
       RMTRIX(I,I) = CANG
       RMTRIX(I,J) = SANG
       RMTRIX(J,I) = -SANG
       RMTRIX(J,J) = CANG
       RETURN
       END
```

Subroutine: ROTEQ

Purpose: ROTEQ evaluates elements of the rotation matrix which relate the general precession of the earth's equator and the consequent retrograde motion of the equinox on the ecliptic. It is used to provide the transformation from mean equator and equinox of 1950.0 to mean equator and equinox of date.

Calling Sequence:

CALL ROTEQ (TIME, A)

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	TIME	1	T	days	Total number days from reference
					epoch (1950.0)
O	A	3,3	A		Rotation matrix

Common storages used or required: None

Subroutines required: None

Functions required: None

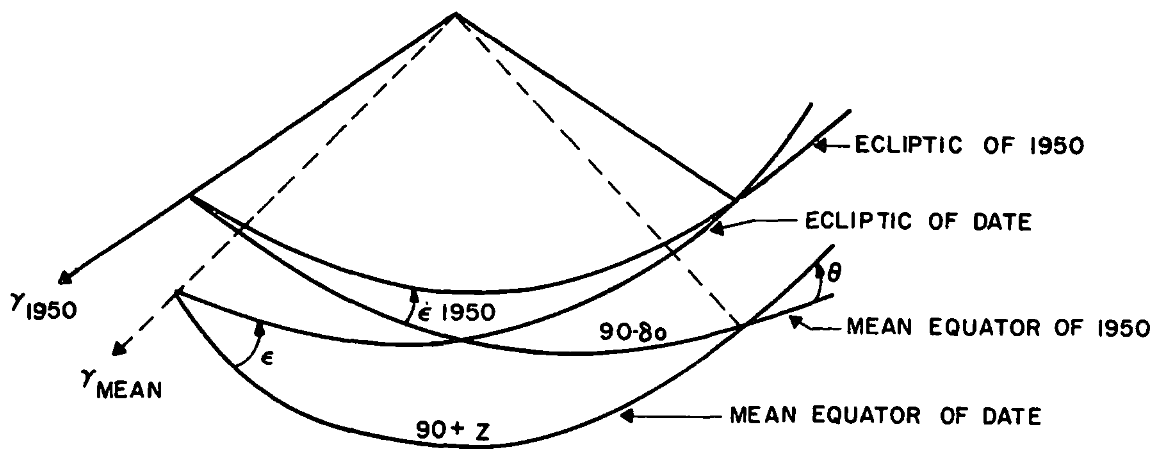
Approximate number of storages required:

Elements of Transformation

The rotation matrix may be represented as

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}$$

where X, Y, Z are expressed in the mean equator and equinox of 1950.0 and x', y', z' are the coordinates in the mean equator and equinox of date.



The geometry of the precession has been represented by the three small parameters δ_o, z, and θ. Υ_{1950} is the mean equinox of 1950; $\bar{\epsilon}_{1950}$ is the mean obliquity of 1950; Υ_{mean} is the mean equinox of date; $\bar{\epsilon}$ is the mean obliquity of date. Measured in the mean equator of 1950 from the mean equinox of 1950, $90^{\circ} - \delta_o$ is the right ascension of the ascending node of the mean equator of date on the mean equator of 1950.

$90^\circ + z$ is the right ascension of the node measured in the mean equator of date from the mean equinox of date. θ is the inclination of the mean equator of date to the mean equator of 1950.

In terms of δ_o, z, and θ, (a_{ij}) is given by:

$$a_{11} = -\sin \delta_o \sin z + \cos \delta_o \cos z \cos \theta$$

$$a_{12} = -\cos \delta_o \sin z - \sin \delta_o \cos z \cos \theta$$

$$a_{13} = -\cos z \sin \theta$$

$$a_{21} = \sin \delta_o \cos z + \cos \delta_o \sin z \cos \theta$$

$$a_{22} = \cos \delta_o \cos z - \sin \delta_o \sin z \cos \theta$$

$$a_{23} = -\sin z \sin \theta$$

$$a_{31} = \cos \delta_o \sin \theta$$

$$a_{32} = - \sin \delta_o \sin \theta$$

$$a_{33} = \cos \theta$$

$$\delta_o = 2304''.997T + ''.302T^2 + ''.0179T^3$$

$$z = 2304''.997T + 1''.093T^2 + ''.0192T^3$$

$$\theta = 2004''.298T - ''.426T^2 - ''.0416T^3$$

with T the number of Julian centries of 36,525 days past the epoch 1950.0.

The actual computational form of (a_{ij}) is obtained by expanding the a_{ij} in power series in δ_o, z, θ and replacing the arguments by the above time series. The results are:

$$a_{11} = 1 - .00029697T^2 - .00000013T^3$$

$$a_{12} = -a_{21} = -.02234988T - .00000676T^2 + .00000221T^3$$

$$a_{13} = -a_{31} = -.00971711T + .00000207T^2 + .00000096T^3$$

$$a_{22} = 1 - .00024976T^2 - .00000015T^3$$

$$a_{23} = a_{32} = -.00010859T^2 - .00000003T^3$$

$$a_{33} = 1 - .00004721T^2 + .00000002T^3$$

```
*       LABEL                                                          ROTQ
*       SYMBOL TABLE                                                   ROTQ
        SUBROUTINE ROTEQ(TIME,A)                                       ROTQ000
        DIMENSION A(3,3)                                               ROTQ001
        T = TIME/36525.                                                ROTQ002
        T2 = T*T                                                       ROTQ003
        T3 = T2*T                                                      ROTQ004
        A(1,1) = 1. - .00029697*T2 - .00000013*T3                      ROTQ005
        A(1,2) = -.02234988*T - .00000676*T2 + .00000221*T3            ROTQ006
        A(2,1) = -A(1,2)                                               ROTQ007
        A(1,3) = -.00971711*T + .00000207*T2 + .00000096*T3            ROTQ008
        A(3,1) = -A(1,3)                                               ROTQ009
        A(2,2) = 1. - .00024976*T2 - .00000015*T3                      ROTQ010
        A(2,3) = -.00010859*T2 - .00000003*T3                          ROTQ011
        A(3,2) = A(2,3)                                                ROTQ012
        A(3,3) = 1. - .00004721*T2 + .00000002*T3                      ROTQ013
        RETURN                                                         ROTQ014
        END                                                            ROTQ
```

Page Intentionally Left Blank

Subroutine: RVIN

Purpose: To transform inertial equatorial spherical to Cartesian coordinates.

Calling Sequence:

CALL RVIN (A, B, C)

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	A	5	A(1) = R		Magnitude of Position Vector
			A(2) = Φ	deg	Declination
			A(3) = θ	deg	Right Ascension
			A(4) = V		Magnitude of Velocity Vector
			A(5) = Γ	deg	Flight Path Angle
			A(6) = Σ	deg	Azimuth
0	B	3			X, Y, Z
0	C	3			$\dot{X}$, $\dot{Y}$, $\dot{Z}$

Common storages used or required: None

Subroutines required: None

Functions required: SIN, COS

Approximate number of storages required:

RVIN

The calling sequence consists of

CALL RVIN (XIN, XOUT, VOUT)

where: XIN is a vector with dimension six. The vector contains the spherical set in the following order

$$XIN = \begin{bmatrix} |R| \\ \phi \\ \theta \\ |V| \\ \gamma \\ \sigma \end{bmatrix}_{6x1} \qquad \begin{aligned} |R| &= \text{Magnitude position vector} \\ \phi &= \text{Declination} \\ \theta &= \text{Right ascension} \\ |V| &= \text{Magnitude velocity vector} \\ \gamma &= \text{Flight path angle} \\ \sigma &= \text{Azimuth} \end{aligned}$$

XOUT is a vector with dimension three. The vector contains the Cartesian components of position

$$XOUT = \begin{bmatrix} x \\ y \\ z \end{bmatrix}_{3x1}$$

VOUT is a vector with dimension three. The vector contains the Cartesian components of velocity

$$VOUT = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix}_{3x1}$$

RVIN TRANSFORMATIONS

The position transformation is obtained as follows (see Fig. 1 below).

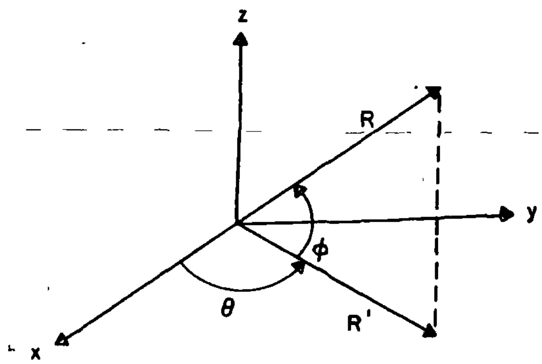


FIG 1

Projecting $\vec{R}$ on the x-y plane, θ is the angle from the x axis to the projection measured counterclockwise. The elevation of $\vec{R}$ above the x-y plane is the angle Φ. The formulas are

$$\text{XOUT} = \underset{3\text{x}1}{\begin{pmatrix} x \\ y \\ z \end{pmatrix}} = |R| \underset{3\text{x}1}{\begin{pmatrix} \cos\Phi\cos\theta \\ \cos\Phi\sin\theta \\ \sin\Phi \end{pmatrix}} \tag{1}$$

The velocity transformation is obtained as follows. The azimuth and flight path angles refer to the tangent plane which is normal to the $\vec{R}$ vector (see Fig. 2).

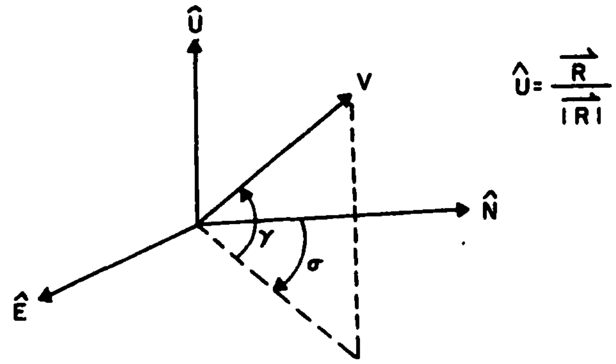


FIG. 2

The unit up, $\hat{U}$, east $\hat{E}$, and north, $\hat{N}$, vectors in terms of xyz are

$$\underset{3x1}{\hat{U}} = \underset{3x1}{\frac{\vec{R}}{|\vec{R}|}} = \underset{3x1}{\begin{pmatrix} \cos\Phi\cos\theta \\ \cos\Phi\sin\theta \\ \sin\Phi \end{pmatrix}} \tag{2}$$

$$\underset{3x1}{\hat{E}} = \underset{3x1}{\frac{\hat{k} \times \hat{U}}{|k \times \hat{U}|}} = \underset{3x1}{\begin{pmatrix} -\sin\theta \\ \cos\theta \\ \Phi \end{pmatrix}} \tag{3}$$

$$\underset{3x1}{\hat{N}} = \underset{3x1}{\hat{U} \times \hat{E}} = \underset{3x1}{\begin{pmatrix} -\sin\Phi\cos\theta \\ -\sin\Phi\sin\theta \\ \cos\Phi \end{pmatrix}} \tag{4}$$

The above vectors may be written in matrix form to represent the transformation from the local tangent plane to the xyz coordinate system

$$\underset{3x1}{\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix}} = \underset{3x3}{\begin{pmatrix} \hat{U} & \hat{E} & \hat{N} \end{pmatrix}} \underset{3x1}{\begin{pmatrix} \dot{U} \\ \dot{E} \\ \dot{N} \end{pmatrix}} = \underset{3x3}{\begin{pmatrix} \cos\Phi\cos\theta & \sin\theta & -\sin\Phi\cos\theta \\ \cos\Phi\sin\theta & \cos\theta & -\sin\Phi\sin\theta \\ \sin\Phi & 0 & \cos\Phi \end{pmatrix}} \underset{3x1}{\begin{pmatrix} \dot{U} \\ \dot{E} \\ \dot{N} \end{pmatrix}} \quad (5)$$

The velocity vector in the tangent plane coordinates (Fig. 2) is

$$\vec{V} = \underset{3x1}{\begin{pmatrix} \dot{U} \\ \dot{E} \\ \dot{N} \end{pmatrix}} = |V| \underset{3x1}{\begin{pmatrix} \sin\gamma \\ \cos\gamma \quad \sin\sigma \\ \cos\gamma \quad \cos\sigma \end{pmatrix}} \quad (6)$$

Substituting equation (6) into (5) yields the desired transformation

$$\text{VOUT} = \underset{3x1}{\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix}} = |V| \underset{3x3}{\begin{pmatrix} \hat{U} & \hat{E} & \hat{N} \end{pmatrix}} \underset{3x1}{\begin{pmatrix} \sin\gamma \\ \cos\gamma \quad \sin\sigma \\ \cos\gamma \quad \cos\sigma \end{pmatrix}}$$

```
*     LABEL                                                    RVIN
*     SYMBOL TABLE                                             RVIN
      SUBROUTINE RVIN(A,B,C)                                   RVIN0000
      DIMENSION A(6),B(3),C(3),D(3,3),E(3)                     RVIN0010
      A(2) = A(2)*.017453293                                   RVIN0020
      A(3) = A(3)*.017453293                                   RVIN0030
      A(5) = A(5)*.017453293                                   RVIN0040
      A(6) = A(6)*.017453293                                   RVIN0050
      CP = COSF(A(2))                                          RVIN0060
      SP = SINF(A(2))                                          RVIN0070
      CT = COSF(A(3))                                          RVIN0080
      ST = SINF(A(3))                                          RVIN0090
      CG = COSF(A(5))                                          RVIN0100
      SG = SINF(A(5))                                          RVIN0110
      CS = COSF(A(6))                                          RVIN0120
      SS = SINF(A(6))                                          RVIN0130
      SPCT = SP*CT                                             RVIN0140
      SPST = SP*ST                                             RVIN0150
      CGSS = CG*SS                                             RVIN0160
      CGCS = CG*CS                                             RVIN0170
      CPCT = CP*CT                                             RVIN0180
      CPST = CP*ST                                             RVIN0190
      D(1,1) = CPCT                                            RVIN0200
      D(1,2) = -ST                                             RVIN0210
      D(1,3) = -SPCT                                           RVIN0220
      D(2,1) = CPST                                            RVIN0230
      D(2,2) = CT                                              RVIN0240
      D(2,3) = -SPST                                           RVIN0250
      D(3,1) = SP                                              RVIN0260
      D(3,2) = 0.                                              RVIN0270
      D(3,3) = CP                                              RVIN0280
      B(1) = A(1)*CPCT                                         RVIN0290
      B(2) = A(1)*CPST                                         RVIN0300
      B(3) = A(1)*SP                                           RVIN0310
      E(1) = A(4)*SG                                           RVIN0320
      E(2) = A(4)*CGSS                                         RVIN0330
      E(3) = A(4)*CGCS                                         RVIN0340
      DO 1 I=1,3                                               RVIN0350
      C(I) = 0.                                                RVIN0360
      DO 1 J=1,3                                               RVIN0370
    1 C(I) = C(I) + D(I,J)*E(J)                                RVIN0380
      RETURN                                                   RVIN0390
      END                                                      RVIN
```

Subroutine: RVOUT

Purpose: To convert Cartesian position and velocity coordinates to inertial equatorial spherical coordinates.

Calling Sequence:

CALL RVOUT (A, B, C)

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	A	3	X,Y,Z		Position coordinates
I	B	3	$\dot{X},\dot{Y},\dot{Z}$		Velocity coordinates
O	C	6	C(1) = R		Magnitude of position vector
			C(2) = Φ	degrees	Declination
			C(3) = θ	degrees	Right ascension
			C(4) = V		Magnitude of velocity vector
			C(5) = Γ	degrees	Flight path angle
			C(6) = Σ	degrees	Azimuth

Common storages used or required: None

Subroutines required: CORSS

Functions required: ATANF, SQRTF, ARKTNS, DOT, ACOS, FNORM

Approximate number of storages required:

RVOUT

The transformations required to convert Cartesian coordinates to inertial equatorial spherical are the inverse of those required by, and described under, RVIN.

For the position coordinates:

$$R = \sqrt{X^2 + Y^2 + Z^2}$$

$$\Phi = \sin^{-1} \frac{Z}{R} \qquad -90^\circ \le \Phi \le 90^\circ$$

$$\theta = \tan^{-1} \frac{Y}{X} \quad \left.\begin{matrix} + 0 \text{ for } X > 0 \\ + 180^\circ \text{ for } X \le 0 \end{matrix}\right\}$$

For the velocity coordinates, $\dot{X}'$, $\dot{Y}'$, $\dot{Z}'$ are obtained by applying the inverse of the transformation matrix $[\hat{U}\ \hat{E}\ \hat{N}]_{3x3}$ given in RVIN. Then:

$$|V| = \sqrt{\dot{X}^2 + \dot{Y}^2 + \dot{Z}^2}$$

$$\Gamma = \sin^{-1} \frac{\dot{X}'}{V} \qquad -90^\circ \le \Gamma \le 90^\circ$$

$$\Sigma = \tan^{-1} \frac{\dot{Y}'}{\dot{Z}'} \quad \left.\begin{matrix} + 0 \text{ for } \dot{Z}' > 0 \\ + 180 \text{ for } \dot{Z}' \le 0 \end{matrix}\right\}$$

```
*     LABEL                                                     RVOT
*     SYMBOL TABLE                                              RVOT
CEC2017  SUBROUTINE RVOUT                                       RVOT
      SUBROUTINE RVOUT(A,B,C)                                   RVOT000
      DIMENSION A(3),B(3), C(6),E(3),G(3),H(3)                  RVOT001
      ACOSF(X) = ATANF(SQRTF(1.-X**2)/X)                        RVOT002
      C(1) = FNORM(A)                                           RVOT003
      C(4) = FNORM(B)                                           RVOT004
      C(2) = ATANF(A(3)/SQRTF(A(1)**2+A(2)**2))                 RVOT005
      C(3)=ARKTNS(360,A(1),A(2))                                RVOT006
    4 F = DOT(A,B)                                              RVOT007
      ARG = F/(C(1)*C(4))                                       RVOT008
      C(5) = ACOSF(ARG)                                         RVOT009
      IF (ARG) 5,6,6                                            RVOT010
    5 C(5) = -1.5707963 - C(5)                                  RVOT011
      GO TO 7                                                   RVOT012
    6 C(5) = 1.5707963 - C(5)                                   RVOT013
    7 CALL CROSS(A,B,E)                                         RVOT014
      EN=FNORM(E)                                               RVOT015
      DO 12 I=1,3                                               RVOT016
   12 E(I)=E(I)/EN                                              RVOT017
      CALL CROSS(E,A,G)                                         RVOT018
      GN = FNORM(G)                                             RVOT019
      DO 8 I=1,3                                                RVOT020
    8 G(I) = G(I)/GN                                            RVOT021
      H(1) = 0.                                                 RVOT022
      H(2) = 0.                                                 RVOT023
      H(3) = 1.                                                 RVOT024
      CALL CROSS(H,A,E)                                         RVOT025
      ARG = DOT(E,G)                                            RVOT026
      EN = FNORM(E)                                             RVOT027
      ARG = ARG/EN                                              RVOT028
      C(6) = ACOSF(ARG)                                         RVOT029
      IF (G(3)) 9,10,10                                         RVOT030
    9 C(6) = C(6) + 1.5707963                                   RVOT031
      GO TO 11                                                  RVOT032
   10 C(6) = -C(6) + 1.5707963                                  RVOT033
   11 CONTINUE                                                  RVOT034
      C(2) = C(2)*57.29578                                      RVOT035
      C(3) = C(3)*57.29578                                      RVOT036
      C(5) = C(5)*57.29578                                      RVOT037
      C(6) = C(6)*57.29578                                      RVOT038
      RETURN                                                    RVOT039
      END                                                       RVOT
```

Page Intentionally Left Blank

Subroutine: SDEC

Purpose: To compute the total perturbation accelerations due to earth's oblateness, perturbing bodies, and Encke perturbations due to deviations from reference conic. These perturbations are obtained by calling OBLN, BODY, and ENCKE and placed in the T block for use in the integration package.

Calling Sequence:

CALL ______

Input and **Output**

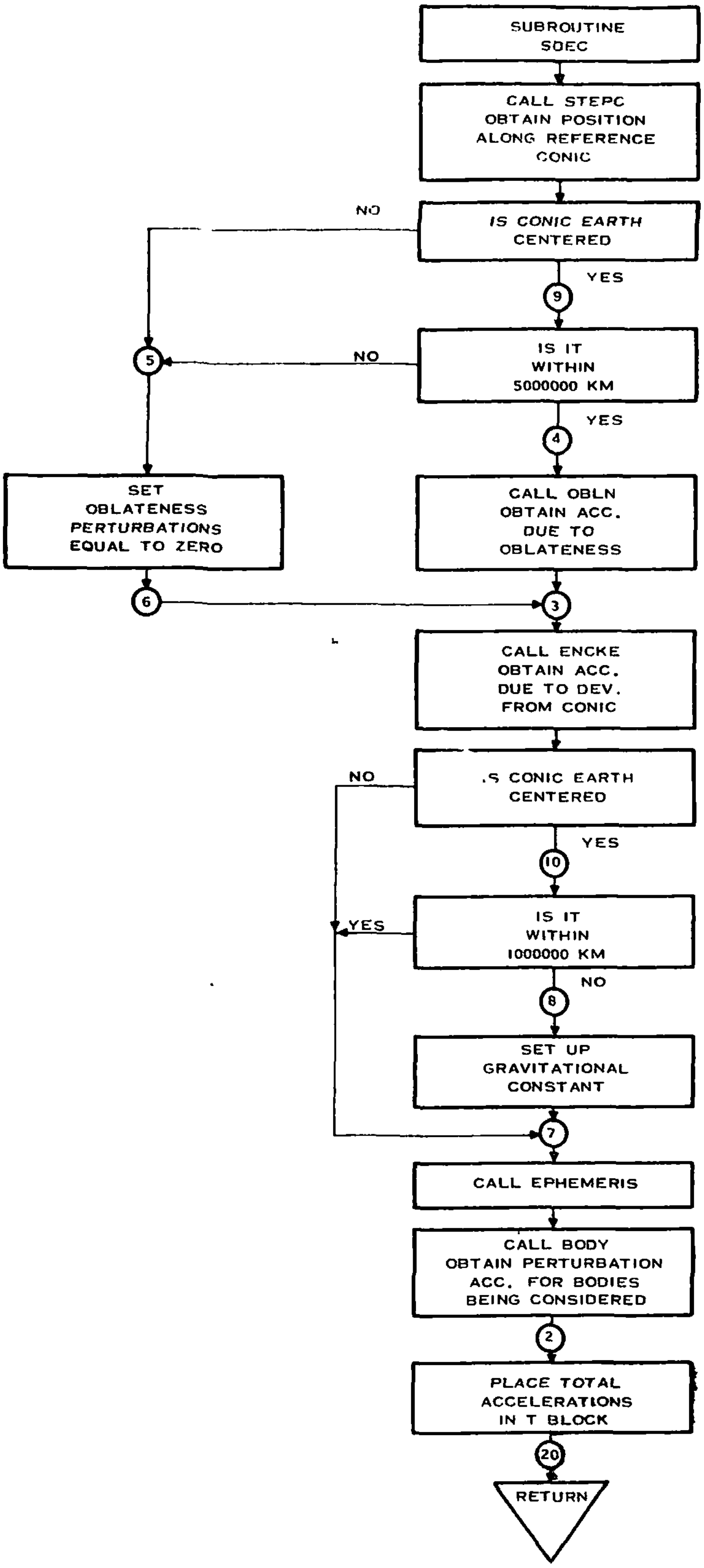
I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I					Quantities obtained from common
O					Quantities placed in common

Common storages used or required: T, S, C, IC

Subroutines required: BODY, ENCKE, INTR, OBLN, STEPC

Functions required: None

Approximate number of storages required: 195 DEC


SUBROUTINE SDEC
CALL STEPC OBTAIN POSITION ALONG REFERENCE CONIC
IS CONIC EARTH CENTERED
NO
YES
9
IS IT WITHIN 5000000 KM
NO
YES
5
4
SET OBLATENESS PERTURBATIONS EQUAL TO ZERO
CALL OBLN OBTAIN ACC. DUE TO OBLATENESS
6
3
CALL ENCKE OBTAIN ACC. DUE TO DEV. FROM CONIC
IS CONIC EARTH CENTERED
NO
YES
10
IS IT WITHIN 1000000 KM
YES
NO
8
SET UP GRAVITATIONAL CONSTANT
7
CALL EPHEMERIS
CALL BODY OBTAIN PERTURBATION ACC. FOR BODIES BEING CONSIDERED
2
PLACE TOTAL ACCELERATIONS IN T BLOCK
20
RETURN

```
*      LABEL                                                            SDEC
*      SYMBOL TABLE                                                     SDEC
CEC2024  SUBROUTINE SDEC                                                SDEC
       SUBROUTINE SDEC                                                  SDEC000
       DIMENSION T(1360),S(1000),C(1000),IC(1)                          SDEC001
       DIMENSION RAD(3),                        X(3),VX(3),XP(3),       SDEC002
     1 AE(3),AO(3),AP(3),PO(22),VE(22),CA(3),VKB(6),                    SDEC003
     2 RBO(6),RBOP(6)          ,UM(6),VXP(3),AN(3,3),EN(3,3),EA(3,3)    SDEC004
     3,BP(3,3)     ,XSO(3),VXSO(3)                                      SDEC005
       COMMON T,S,C,IC                                                  SDEC006
       EQUIVALENCE (T,TDUM),(S,SDUM),(C,CDUM)                           SDEC007
       EQUIVALENCE (S(11),VJ),(S(12),H),(S(13),D),(S(900),RAD)          SDEC008
       EQUIVALENCE (C(10),TW),(C(11),TF),(C(13),TP1),(C(14),TP2)        SDEC009
     1,(C(15),XP),(C(18),VXP),(C(21),AE),(C(24),AO),(C(27),CA)          SDEC010
     2,(C(33),X),(C(36),VX)                                             SDEC011
     4,(C(55),U),(C(56),VKB),(C(62),PO),(C(84),VE),(C(106),RBO)         SDEC012
     5,(C(112),RBOP),(C(120),EN),(C(129),EA),(C(138),AN),(C(171),BP)    SDEC013
     7,(IC(1),N),(IC(2),IOR),(IC(3),NOR),(IC(4),KSTP),(IC(5),KTYPE)     SDEC014
     8,(IC(6),ITARG),(S(14),RE),(C(46),XSO),(C(49),VXSO),(IC(231),NSTEP)SDEC015
       NEQ=3*N                                                          SDEC016
       CALL STEPC(NSTEP,T(2),XSO,VXSO,U,X,VX)                           SDEC017
       DO 1 I=1,3                                                       SDEC018
     1 XP(I) = X(I) + T(I+3)                                            SDEC019
       IF(NOR-1)5,9,5                                                   SDEC020
     9 IF (RBOP-500000.) 4,5,5                                          SDEC021
     4 CALL OBLN(XP,U,RE,VJ,H,D,AO,AN,BP,NEQ)                           SDEC022
       GO TO 3                                                          SDEC023
     5 DO 6 I=1,3                                                       SDEC024
       AO(I)=0.                                                         SDEC025
       DO 6 J=1,3                                                       SDEC026
       BP(I,J)=0.                                                       SDEC027
     6 CONTINUE                                                         SDEC028
     3 CALL ENCKE(U,X,T(4),AE)                                          SDEC029
       IF(NOR-1)7,10,7                                                  SDEC030
    10 IF (RBOP-1000000.) 7,8,8                                         SDEC031
     8 VKB(6) = UM(6)                                                   SDEC032
     7 NV = 0                                                           SDEC033
       NR1=NOR-1                                                        SDEC034
       TP2 = TF+T(2)/86400.                                             SDEC035
       TWGN=0.                                                          SDEC036
    21 CONTINUE                                                         SDEC037
       IF(TP2)22,23,23                                                  SDEC038
    22 CONTINUE                                                         SDEC039
       TP2=TP2+1.                                                       SDEC040
       TWGN=TWGN-1.                                                     SDEC041
       GO TO 21                                                         SDEC042
    23 CONTINUE                                                         SDEC043
       TFP = INTF(TP2)                                                  SDEC044
       TP2 = TP2-TFP                                                    SDEC045
       TP1=TW+TFP+TWGN                                                  SDEC046
       CALL INTR(TP1,TP2,NR1,PO,NV,VE,RBOP)                             SDEC047
```

```
      CALL BODY(XP,PO,VKB,RBO,RBOP,CA,NOR,BP,T,NEQ,U)               SDEC0480
      KJ=2*NEQ+3                                                    SDEC0490
      DO 2 I=1,3                                                    SDEC0500
      KI=KJ+I                                                       SDEC0510
    2 T(KI) = AO(I) + AE(I) -CA(I)                                  SDEC0520
   20 RETURN                                                        SDEC0530
      END                                                           SDEC
```

Subroutine: SETN

Purpose: SETN sets the logical tape numbers for input and output to conform with the system unit table.

Calling Sequence:

CALL SETN (NIN, NOUT)

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
O	NIN	1			Logical input tape unit
O	NOUT	1			Logical output tape unit

Common storages used or required: None

Subroutines required: None

Functions required: None

Approximate number of storages required:

```
*       LABEL                                                        SETN
CEC2023                                                              SETN0000
        SUBROUTINE SETN(NIN,NOUT)                                    SETN0010
C       CALLING THIS SUBROUTINE RESULTS IN THE USE OF                SETN0020
C       INPUT TAPE 2 AND OUTPUT TAPE 3                               SETN0030
C       USER MUST READ INPUT TAPE NIN,  AND WRITE OUTPUT NOUT,       SETN0040
        NIN=2                                                        SETN0050
        NOUT=3                                                       SETN0060
        RETURN                                                       SETN0070
        END                                                          SETN
```

Subroutine: SHIFTP

Purpose: To change position and velocity coordinates from a coordinate system centered at body IOR to one centered at body NOR. SHIFTP also returns the new gravitational constant and the position and velocity of all bodies on the ephemeris tape referenced to the new center. (The latter is done by calling the ephemeris subroutine INTRI)

Calling Sequence:

CALL SHIFTP (IOR, NOR, U, UM, X, VX, PO, VE, TP1, TP2)

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	IOR	1			Index of old central body
I	NOR	1			Index of new central body
O	U	1	μ	Km^3/sec^2	Gravitational constant, body NOR
I	UM	6			Array of gravitational constants
I-O	X	3	X, Y, Z	Km	Positions input: * w.r.t. body IOR
I-O	VX	3	$\dot{X}, \dot{Y}, \dot{Z}$	Km/sec	Velocities output: w.r.t. body NOR
O	PO	22		Km	Positions of bodies on tape w.r.t.
O	VE	22		Km/sec	Velocities body NOR
I	TP1	1		seconds	Double precision time in sec-
I	TP2	1		seconds	onds since 1950.

Common storages used or required: None

Subroutines required: INTRI

Functions required: None

Approximate number of storages required: ________

* with respect to

The sequence used in shifting centers is the following:

IOR (NOR)	CENTRAL BODY
1	Earth
2	Moon
3	Sun
4	Venus
5	Mars
6	Jupiter

```
*     LABEL                                                              SHTP
*     SYMBOL TABLE                                                       SHTP
      SUBROUTINE SHIFTP(IOR,NOR,U,UM,X,VX,PO,VE,TF,TW,T2)                SHTP000
      COMMON T,S,C,IC                                                    SHTP001
      DIMENSION T(1360),S(1000),C(1000),IC(1)                            SHTP002
      DIMENSION UM(6),X(3),VX(3),PO(22),VE(22)                           SHTP003
      EQUIVALENCE (T,TDUM),(S,SDUM),(C,CDUM),(C(14),TP2),(C(13),TP1)     SHTP004
      N = NOR                                                            SHTP005
      IO = IOR                                                           SHTP006
      IOR = NOR                                                          SHTP007
      U = UM(N)                                                          SHTP008
      NV = 1                                                             SHTP009
      DIS = 1.E10                                                        SHTP010
      NR1 = NOR-1                                                        SHTP011
      CALL INTR1(TP1,TP2,NR1,PO,NV,VE,DIS)                               SHTP012
      DO 1 I=1,3                                                         SHTP013
      J = 26-3*IO-I                                                      SHTP014
      X(I) = X(I)+PO(J)                                                  SHTP015
    1 VX(I) = VX(I)+VE(J)                                                SHTP016
      RETURN                                                             SHTP017
      END                                                                SHTP
```

Page intentionally left blank

Page intentionally left blank

Subroutine: STEPC

Purpose: To obtain the state, i.e., radius vector, R, and velocity vector, V, of a probe which is separated by a time increment, t, from a specified state, R_o and V_o, assuming an inverse-square central force law with gravitational constant u.

Calling Sequence:

CALL STEPC (N, T, R, V, U, RR, VV)

Input and **Output**

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	N	1			Initializing index
I	T	1	Δt	sec	Incremental time
I	R	3	$\vec{X}_o$	Km	Initial Conic position (t_o)
I	V	3	$\dot{\vec{X}}_o$	Km/sec	Initial Conic Velocity (t_o)
I	U	1	μ	Km^3/sec^2	Gravitational constant
O	RR	3	$\vec{X}$	Km	Position on Conic ($t_o + \Delta t$)
O	VV	3	$\dot{\vec{X}}$	Km/sec	Velocity on Conic ($t_o + \Delta t$)

Common storages used or required: None

Subroutines required: DOT, GOTOR

Functions required: ARKTNS, SQRT

Approximate number of storages required: 255 DEC

*Discussion

The inverse-square central force law is characterized by the equations

$$\dot{R} = \frac{dR}{dt} = V$$

$$\dot{V} = \frac{dV}{dt} = -\frac{u}{r^3} R \tag{1}$$

where r is defined to be $+(R \cdot R)^{1/2}$.

The angular momentum vector, H, defined by the vector cross-product

$$H = R \times V \tag{2}$$

is a constant with respect to time, as seen by

$$\dot{H} = \dot{R} x V + R x \dot{V} = V x V + R x \left(\frac{-u}{r^3} R \right) = 0, \text{ so that } H = H_o = R_o \times V_o.$$

Because the magnitude, h_o, of H_o is generally nonzero, the plane equations

$$R \cdot H = R \cdot H_o = 0$$

$$V \cdot H = V \cdot H_o = 0 \tag{3}$$

which follow from the definition of H, lead to the conclusion that all motion occurs in the plane normal to H_o. That is, if R_o and V_o are nonzero and noncollinear vectors, any other vectors, such as R and V, must lie in the plane formed by R_o and V_o. Algebraically, one says that R and V may be expressed as linear combinations of R_o and V_o.

$$R = f\, R_o + g\, V_o$$

$$V = \dot{f}\, R_o + \dot{g}\, V_o \tag{4}$$

* Vectors will be denoted by capital letters and scalars by lower-case letters with the single exception of the rather standard symbol, E, for eccentric anomaly.

The second of equations (4) follows from (1) and from the fact that R_o and V_o are not functions of the time increment, t. The scalars, f, $\dot{f}$, g, and $\dot{g}$, are functions of R_o, V_o, and t. The discussion which follows concerns the determination of these scalars.

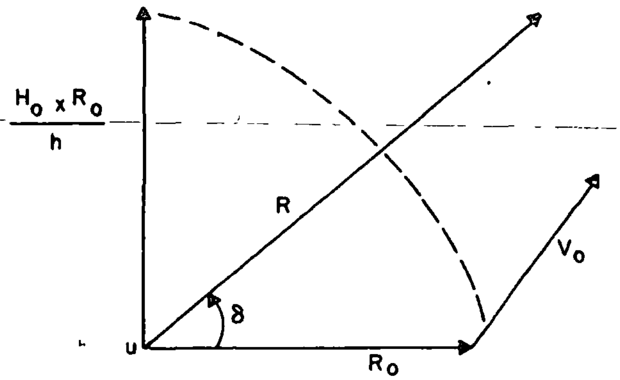


FIG 1

The vector $(H_o \times R_o)/h$ is seen to be perpendicular to R_o and to have the same magnitude, r_o, as R_o. Figure 1 shows that R may be written as the linear combination,

$$R = \frac{r}{r_o}\left[R_o \cos\delta + \frac{H_o \times R_o}{h} \sin\delta\right] \tag{5}$$

Equation (5) may be rewritten by expanding the vector triple cross-product,

$$H_o \times R_o = (R_o \times V_o) \times R_o = V_o\,(R_o \cdot R_o) - R_o\,(R_o \cdot V_o).$$

$$R = \frac{r}{r_o}\left[R_o\left(\cos\delta - \frac{R_o \cdot V_o}{h}\sin\delta\right) + V_o\,\frac{r_o^2}{h}\sin\delta\right] \tag{5a}$$

Comparing equation (5a) with equation (4),

$$f = \frac{r}{r_o}\left(\cos\delta - \frac{R_o\cdot V_o}{h}\ \sin\delta\right)$$

$$g = \frac{rr_o}{h}\ \sin\delta. \tag{6}$$

At this point, the reader should note that while r_o, $R_o\cdot V_o$, and h are computable directly from R_o and V_o, the quantities r and δ have not been specifically defined in terms of R_o, V_o, and t. In order to do so, it is necessary to examine the conic solution to the inverse-square law equations of motion.

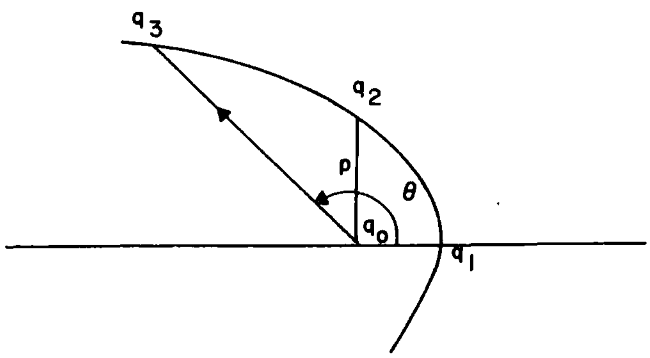


FIG 2

Figure 2 shows a conic section with <u>focus</u> at q_o and <u>pericenter</u> at q_1. The point q_3 on the conic is specified, in polar coordinates originating at q_o, by the <u>radius</u>, r, and <u>true anomaly</u>, θ, θ being measured positive counter-clockwise from the line $\overline{q_oq_1}$. The length $\overline{q_oq_2}$ is called the <u>semi-latus rectum</u> and is denoted by p. The parameter specifying the shape of the conic is the <u>eccentricity</u>, e, which may be calculated from the equation $e = \frac{\overline{q_oq_2}}{\overline{q_oq_1}} - 1$. The conic equation in an often-used form is,

$$r = \frac{p}{1+e\cos\theta} \tag{7}$$

Conic sections are characterized by eccentricity as follows:

circular	if	$e = 0$
elliptical	if	$0 < e < 1$
parabolic	if	$e = 1$
hyperbolic	if	$e > 1$

The two cases of great interest here are the elliptical and hyperbolic cases. Figure 3 below shows an ellipse.

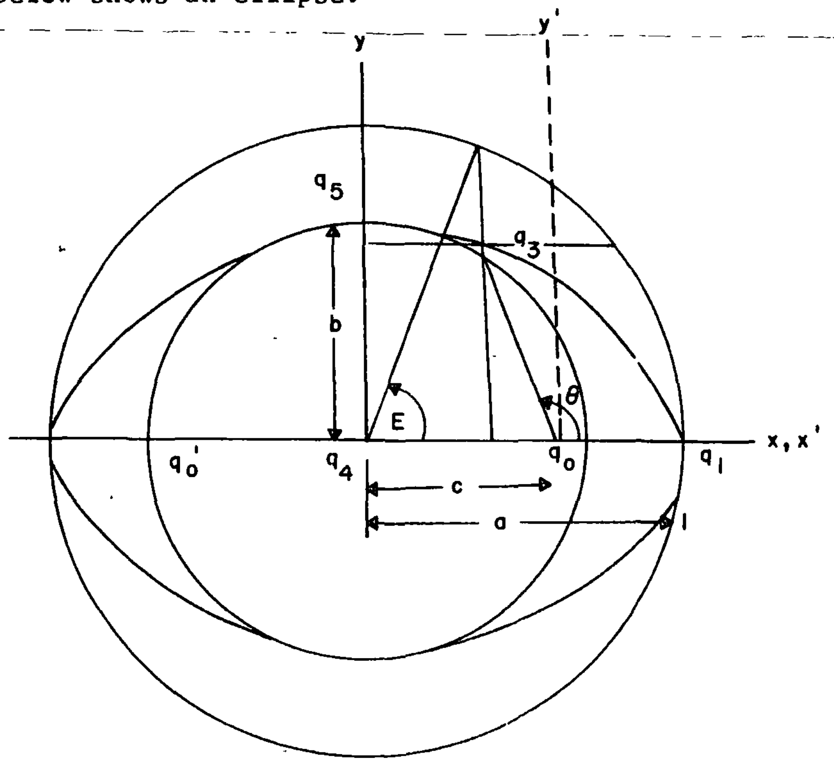


FIG. 3

The semi-major axis, a, is the distance from the point of symmetry, q_4, to the pericenter, q_1. The semi-minor axis, b, is the length $\overline{q_4 q_5}$. The eccentricity is equal to the ratio

$$c/a = \frac{\sqrt{a^2 - b^2}}{a}$$

The eccentric anomaly, E, is the argument of the projection of the point, q_3, from the line $\overline{q_4 q_1}$ onto a circle of radius, a, concentric with the ellipse. That is, the Cartesian coordinates of q_3 relative to q_4 are

$$x = a \cos E$$

$$y = b \sin E \tag{8}$$

Equations (8) give rise to the familiar equation of the ellipse

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = \cos^2 E + \sin^2 E = 1.$$

Relative to coordinates centered at q_o, q_3 is described by

$$x' = r\cos\theta = a\cos E - ea$$

$$y' = y = r\sin\theta = b\sin E = a\sqrt{1-e^2}\sin E. \tag{8a}$$

$$r = x'^2 + y'^2 = a(1-e\cos E)$$

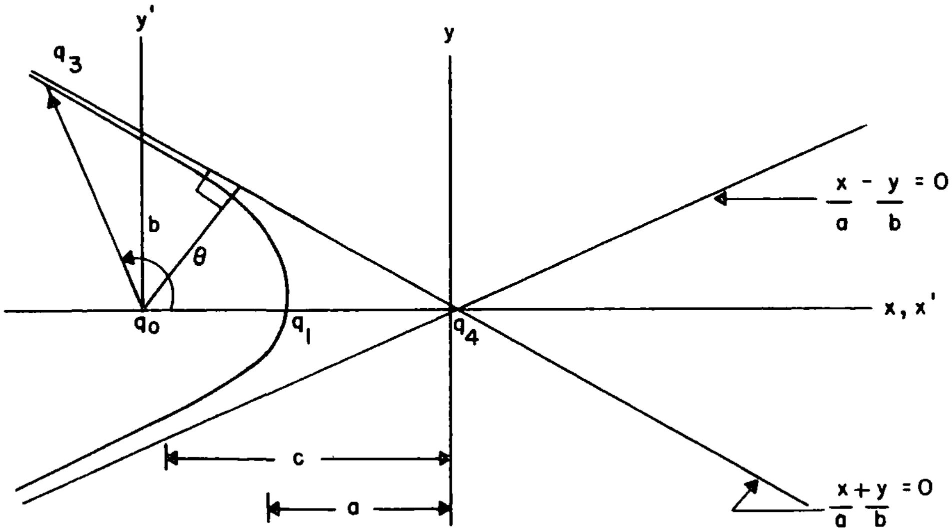


FIG 4

Figure 4 shows a conic section which is hyperbolic. The lines $\frac{x}{a} - \frac{y}{b} = 0$ and $\frac{x}{a} + \frac{y}{b} = 0$ are the <u>asymptotes</u> of the hyperbola. The <u>semi-major axis</u>, a, is again the distance from the point of symmetry, q_4, to the <u>pericenter</u>, q_1. The <u>semi-minor axis</u>, b, is defined as the distance from the <u>focus</u>, q_o, to one of the asymptotes. <u>Eccentricity</u> is given by the ratio

$$c/a = \sqrt{\frac{a^2 + b^2}{a^2}} .$$

The <u>eccentric anomaly</u>, E, (sometimes called F) is too difficult to picture to be included in Fig. 4. E is defined in such a way that q_3 is described in Cartesian coordinates by

$$x = -a \cosh E$$
$$y = b \sinh E, \qquad (9)$$

leading to the familiar hyperbolic equation

$$\frac{x^2}{a^2} - \frac{y^2}{b^2} = \cosh^2 E - \sinh^2 E = 1$$

The Cartesian coordinates of q_3 relative to q_o are given by

$$x' = r\cos\theta = ea - a\cosh E$$
$$y' = r\sin\theta = b\sinh E = a\sqrt{e^2-1}\,\sinh E. \qquad (9a)$$
$$r = \sqrt{x'^2 + y'^2} = a(e\cosh E - 1)$$

Solution of the differential equations (1) yields p, a, and e as constants of the motion.

It is easily verified that

$$h = \sqrt{r_o^2 v_o^2 - (R_o \cdot V_o)^2} = r^2\dot{\theta}$$ <u>angular momentum magnitude</u>

$$p = h^2/u$$ semi-latus rectum

$$a = \frac{r_o}{\left|2 - \frac{r_o v_o^2}{u}\right|}$$ semi-major axis

$$e^2 = 1 - p/a$$ eccentricity

$$n = \sqrt{u/a^3}$$ mean motion

It should be observed that p is non-negative so that a negative semi-major axis corresponds to an eccentricity greater than one; that is, to the hyperbolic case. Some useful identities are listed as equations (11).

$$\frac{R \cdot V}{r} = \frac{d}{dt} (R \cdot R)^{1/2} = \dot{r} = \frac{ep\dot{\theta}\sin\theta}{(1+e\cos\theta)^2} = \frac{ue}{h}\sin\theta$$

$$\cos\theta = \mp \frac{a}{r}(cE - e)$$

$$\sin\theta = \frac{a}{r}\sqrt{\left|e^2 - 1\right|}\, sE = \sqrt{\frac{a}{u}}\,\frac{h}{r}\, sE$$

$$esE = \frac{R \cdot V}{\sqrt{ua}}$$

$$ecE = 1 \pm \frac{r}{a} \qquad (11)$$

The convention has been adopted that when a double sense sign $\pm$ is used, the upper pertains to the hyperbolic case and the lower to the elliptical.

The symbols sE and cE mean sinE and cosE for the elliptical case, but sinhE and coshE for the hyperbolic case. Hence, if we let

$\Phi = E - E_o$, it follows that

$$s\Phi = sEcE_o - cEsE_o$$

$$c\Phi = cEcE_o \mp sEsE_o$$

and

$$sE = s\Phi cE_o + c\Phi sE$$

$$cE = c\Phi cE_o \pm s\Phi cE_o \qquad (12)$$

This angle, δ, of equation (5) is seen to represent the incremental true anomaly on the conic section containing R_o and R. Thus, we can

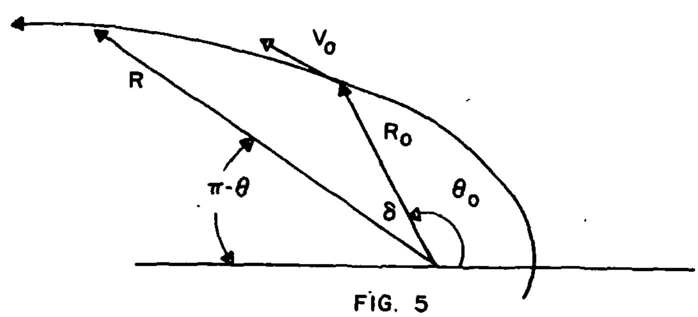


FIG. 5

find expressions for sin δ and cos δ in terms of θ and θ_o.

$$\sin\delta = \sin(\theta-\theta_o) = \sin\theta\cos\theta - \cos\theta\sin\theta_o$$

$$\cos\delta = \cos(\theta-\theta_o) = \cos\theta\cos\theta_o + \sin\theta\sin\theta_o. \qquad (13)$$

The incremental true anomaly may now be related to R_o and V_o through substitution of equations (11) and (12) into (13).

$$\sin\delta = \sin\theta\cos\theta_o - \cos\theta\sin\theta_o$$

$$= \sqrt{\frac{a^3}{u}}\ \frac{h}{rr_o}\ (\mp sEcE_o \pm esE \mp esE_o \pm cEsE_o)$$

$$= \frac{h}{nrr_o}\ (\mp s\Phi \pm esE \mp esE_o)$$

$$= \frac{h}{nrr_o}\ (\mp s\phi \pm ecE_o s\phi \pm esE_o c\phi \mp esE_o)$$

$$= \frac{h}{nrr_o}\left(\frac{r_o}{a}\ s\Phi \pm \frac{R_o \cdot V_o}{\sqrt{ua}}\ (c\Phi - 1)\right)$$

$$\cos\delta = \cos\theta\cos\theta_o + \sin\theta\sin\theta_o$$

$$= \frac{a^2}{rr_o}\left[cEcE_o - ecE - ecE_o + e^2 \pm (e^2-1)\, sEsE_o\right]$$

$$= \frac{a^2}{rr_o}\left[c\Phi - ecE - ecE_o + e^2 \pm e^2 sEsE_o\right]$$

$$= \frac{a^2}{rr_o}\left[c\Phi - ec\Phi cE_o \mp es\Phi sE_o - ecE_o + e^2 \pm e^2 sE_o cE_o s\Phi \pm e^2 s^2 E_o c\Phi\right]$$

$$= \frac{a^2}{rr_o}\left[\mp \frac{r_o}{a}\ c\Phi + \frac{r_o}{a}\ \frac{R_o \cdot V_o}{\sqrt{ua}}\ s\Phi + 1 \pm \frac{p}{a} - 1 \mp \frac{r_o}{a} \pm \frac{(R_o \cdot V_o)^2}{ua}\ c\Phi\right]$$

$$= \frac{a^2}{rr_o}\left[\mp \frac{r_o}{a}\ (c\Phi + 1) + \frac{r_o}{a}\ \frac{R_o \cdot V_o}{\sqrt{ua}}\ s\Phi \pm \frac{r_o^2 v_o^2}{ua} \pm \frac{(R_o \cdot V_o)^2}{ua}\ (c\Phi - 1)\right]$$

$$= \frac{a^2}{rr_o}\left[\mp \frac{r_o}{a}\ (c\Phi - 1) + \frac{r_o}{a}\ \frac{R_o \cdot V_o}{\sqrt{ua}}\ s\Phi + \frac{r_o}{a}\left(\frac{r_o v_o^2}{u} - 2\right)\right.$$
$$\left.\pm \frac{(R_o \cdot V_o)^2}{ua}\ (c\Phi - 1)\right]$$

$$= \frac{a^2}{rr_o}\left[\pm\left[\frac{(R_o \cdot V_o)^2}{ua} - \frac{r_o}{a}\right](c\Phi - 1) + \frac{r_o}{a}\ \frac{R_o \cdot V_o}{\sqrt{ua}}\ s\Phi + \frac{r_o^2}{a^2}\right]$$

The coefficients, f and g, of equation (6) are finally found to be:

$$f = \frac{r}{r_o}\left(\cos\delta - \frac{R_o \cdot V_o}{h}\sin\delta\right)$$

$$= \frac{a^2}{r_o^2}\left[\pm\left[\frac{(R_o \cdot V_o)^2}{ua} - \frac{r_o}{a} - \frac{(R_o \cdot V_o)^2}{ua}\right](c\Phi-1) + \frac{r_o}{a}\left[\frac{R_o \cdot V_o}{\sqrt{ua}} - \frac{R_o \cdot V_o}{\sqrt{ua}}\right]s\Phi\right.$$

$$\left. + \frac{r_o^2}{a^2}\right] = \mp\frac{a}{r_o}(c\Phi-1) + 1 \qquad (14)$$

$$g = \frac{rr_o}{h}\sin\delta$$

$$= \frac{1}{n}\frac{r_o}{a}s\Phi \pm \frac{R_o \cdot V_o}{\sqrt{ua}}(c\Phi-1) \qquad (15)$$

The velocity coefficients, $\dot{f}$ and $\dot{g}$, are found by noting that

$$\dot{\Phi} = \dot{E} = \sqrt{\frac{u}{a}}\,\frac{1}{r}$$

and therefore $\dot{f} = \mp\frac{a}{r_o}\frac{d}{dt}(c\Phi) = -\frac{a}{r_o}\dot{E}s\Phi = -\frac{\sqrt{ua}}{rr_o}s\Phi$ (16)

$$\dot{g} = \frac{\dot{E}}{n}\left(\frac{r_o}{a}c\Phi + \frac{R_o \cdot V_o}{\sqrt{ua}}s\Phi\right)$$

$$= \frac{a}{r}\left(\frac{r_o}{a}c\Phi + \frac{R_o \cdot V_o}{\sqrt{ua}}s\Phi\right) \qquad (17)$$

where

$$\frac{r}{a} = \mp(1-ecE) = \mp(1-ec\Phi cE_o \mp es\Phi sE_o)$$

$$= \mp\left(1-c\Phi\left(1 \pm \frac{r_o}{a}\right) \mp s\Phi\frac{R_o \cdot V_o}{\sqrt{ua}}\right)$$

$$= \pm(c\Phi-1) + \frac{r_o}{a}c\Phi + \frac{R_o \cdot V_o}{\sqrt{ua}}s\Phi \qquad (18)$$

Kepler's equation presents the incremental time, t, in terms of Φ

$$nt = \mp (E-esE) \pm (E_o-esE_o)$$

$$= \mp (\Phi-esE+esE_o)$$

$$= \mp (\Phi-es\Phi cE_o-ec\Phi sE_o+esE_o)$$

$$= \mp \left[\Phi-s\Phi\left(1 \pm \frac{r_o}{a}\right) - (c\Phi-1)\frac{R_o \cdot V_o}{\sqrt{ua}}\right]$$

$$= \mp (\Phi-s\Phi) + \frac{r_o}{a} s\Phi \pm \frac{R_o \cdot V_o}{\sqrt{ua}} (c\Phi-1) \qquad (19)$$

In order to solve equation (19) for Φ as a function of R_o, V_o and t, an iterative method is required. Any solution obtained is necessarily unique, since the right side of (19) is seen to be monotonically increasing for all Φ.

Equation (19) may be used to simplify equations (15) and (17).

$$g = \frac{1}{n}\left[\frac{r_o}{a} s\Phi+nt\pm (\Phi-s\Phi) - \frac{r_o}{a} s\Phi\right]$$

$$= t \pm \frac{1}{n} (\Phi-s\Phi) \qquad (15a)$$

$$\dot{g} = 1 \pm \frac{\dot{E}}{n} (1-c\Phi)$$

$$= 1 \mp \sqrt{\frac{u}{a}} \frac{1}{nr} (c\Phi-1)$$

$$= 1 \mp \frac{a}{r} (c\Phi-1) \qquad (17a)$$

Subroutine Description

The subroutine will calculate

(1) $R = R(R_o, V_o, t)$

$V = V(R_o, V_o, t)$

$\delta = \delta(R_o, V_o, t)$

The first time STEP is called with a given R_o and V_o (N=1), those coefficients which are dependent only on R_o and V_o are calculated. Thereafter, these coefficients are considered to be constants with respect to t.

Calculation	FORTAN Name
$r_o = \sqrt{R_o \cdot R_o}$	RM
$v_o = \sqrt{V_o \cdot V_o}$	VM
$a = \dfrac{r_o}{\left\| 2 - \dfrac{r_o v_o^2}{u} \right\|}$	A
$n = \dfrac{1}{a}\sqrt{\dfrac{u}{a}}$	RAT
$R_o \cdot V_o = R_o \cdot V_o$	RV
$h = \sqrt{r_o^2 v_o^2 - (R_o \cdot V_o)^2}$	H

STEPC calls Subroutine GOTOR to solve equation (19),

$$nt = \mp (\Phi - s\Phi) + \frac{r_o}{a} s\Phi \pm \frac{R_o \cdot V_o}{\sqrt{ua}} (c\Phi - 1) \text{ for } \Phi.$$

GOTOR returns a vector F where

		Elliptical	Hyperbolic
F(1)	=	Φ-sin Φ	sinh Φ-Φ
F(2)	=	1-cos Φ	cosh Φ-1
F(3)	=	sin Φ	sinh Φ
F(4)	=	cos Φ	cosh Φ

The coefficients, f, g, $\dot{f}$ and $\dot{g}$, are then calculated.

Calculation	FORTRAN Name
$f = -\frac{r_o}{a} F(2) + 1$	EF
$g = -\frac{F(1)}{n} + t$	GE
$\frac{r}{a} = F(2) + \frac{r_o}{a} F(4) + \frac{R_o \cdot V_o}{\sqrt{ua}} F(3)$	ROA
$\dot{f} = - anF(3) / \frac{(r_o r)}{a}$	EFD
$\dot{g} = - \frac{a}{r} F(2) + 1$	GED
$\delta = \tan^{-1} \frac{hg}{f_o^2 f + R_o \cdot V_o g}$	TA
$R = fR_o + gV_o$	RR
$V = \dot{f}R_o + \dot{g}V_o$	VV

Reference: NASA X-640-63-71, ITEM Program Manual, Goddard Space Flight Center, Greenbelt, Maryland.

```
      LABEL
*      SYMBOL TABLE
CEC2023  SUBROUTINE STEPC
       SUBROUTINE STEPC(N,T,R,V,U,RR,VV)
       DIMENSION R(3),V(3),RR(3),VV(3),C(2),F(4)
       GO TO (1,2),N
     1 CONTINUE
       PI=3.141592654
       R2=DOT(R,R)
       RM=SQRTF(R2)
       V2=DOT(V,V)
       RV=DOT(R,V)
       VM=SQRTF(V2)
       A=RM/(2.-RM*V2/U)
       ABA=ABSF(A)
       RAT=SQRTF(U/ABA)/ABA
       C(1)=RM/ABA
       C(2)=RV/SQRTF(U*ABA)
       IF(A)3,3,4
     4 CONTINUE
       K=1
       E1=RAT*T
       GO TO 15
     3 CONTINUE
       K=2
       E1=0.
    15 CONTINUE
     2 CONTINUE
C      T IS THE INCREMENTAL TIME IN ORBIT
       EMDT=RAT*T
       CALL GOTOR(K,EMDT,C,F,E1)
       EF=-F(2)/C(1)+1.
       GE=-F(1)/RAT+T
       TA=ARKTNS(360,R2*EF+RV*GE,H*GE)
       ROA=F(2)+C(1)*F(4)+C(2)*F(3)
       EFD=-ABA*RAT*F(3)/(RM*ROA)
       GED=-F(2)/ROA+1.
     8 CONTINUE
       DO 10 I=1,3
       RR(I)=EF*R(I)+GE*V(I)
       VV(I)=EFD*R(I)+GED*V(I)
    10 CONTINUE
       RETURN
       END
```

Page Intentionally Left Blank

Subroutine: TIMEC

Purpose: To compute whole days and fractional days since January 1, 1950, from calendar date input. This input is in the form (Years from 1900) x 10^2 + (Month of Year) + (Day of Month) x 10^{-2}, and (Hour of Day) x 10^2 + (Minutes of Hour) + (Seconds of Minute) x 10^{-2}. [See examples under input description]. The input date must be 1961 or later.

Calling Sequence:

CALL TIMEC (T1, T2, T3, T4)

Input and **Output**

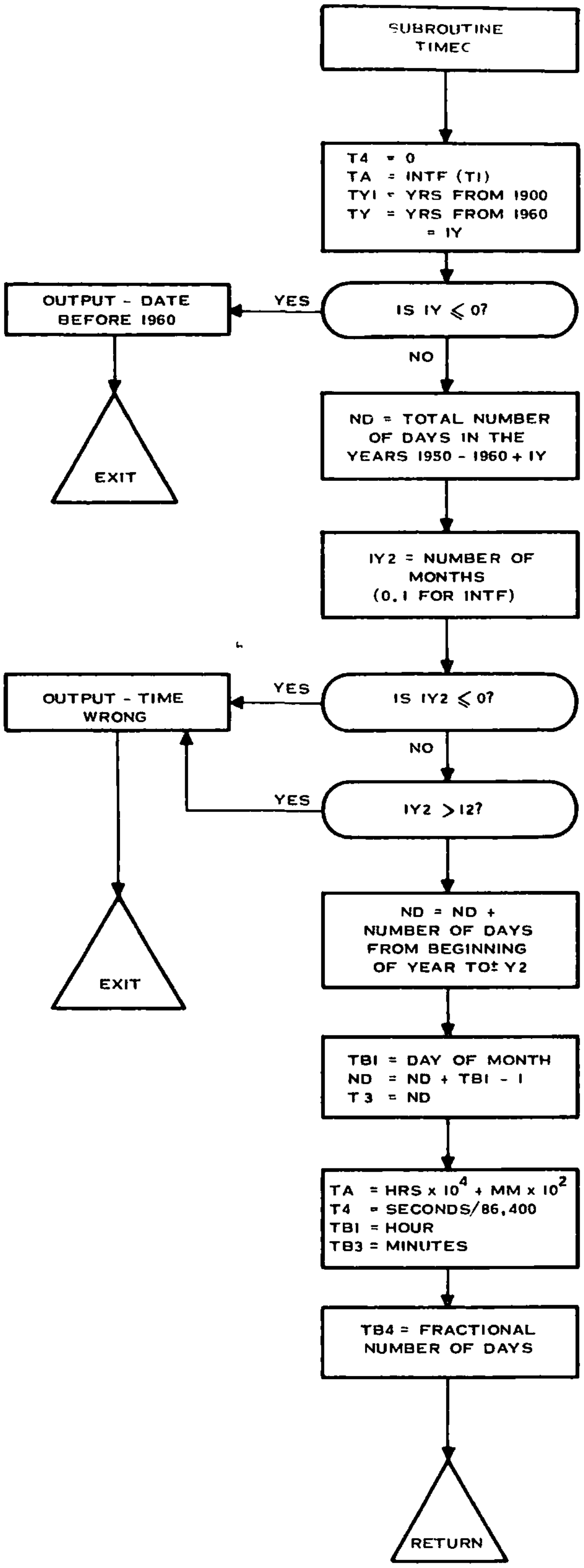
I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	T1	1			Year, month, day.
					Jan. 12, 1965 = 6501.12.
I	T2	1			Hour, minutes, seconds.
					1 PM, 1 min, 30.3 sec = 1301.303
O	T3	1		days	Whole number of days since 1950.
O	T4	1		days	Fractional number of days past
					T3.

Common storages used or required: None

Subroutines required: EXIT

Functions required: INTF

Approximate number of storages required:


SUBROUTINE TIMEC
T4 = 0
TA = INTF (TI)
TYI = YRS FROM 1900
TY = YRS FROM 1960 = IY
IS IY ⩽ 0?
YES
OUTPUT - DATE BEFORE 1960
EXIT
NO
ND = TOTAL NUMBER OF DAYS IN THE YEARS 1950 - 1960 + IY
IY2 = NUMBER OF MONTHS (0.1 FOR INTF)
IS IY2 ⩽ 0?
YES
OUTPUT - TIME WRONG
EXIT
NO
IY2 > 12?
YES
ND = ND + NUMBER OF DAYS FROM BEGINNING OF YEAR TO IY2
TBI = DAY OF MONTH
ND = ND + TBI - I
T3 = ND
TA = HRS x 10^4 + MM x 10^2
T4 = SECONDS/86,400
TBI = HOUR
TB3 = MINUTES
TB4 = FRACTIONAL NUMBER OF DAYS
RETURN

```
*      LABEL
*      SYMBOL TABLE
       SUBROUTINE TIMEC(T1,T2,T3,T4)
C      T1 IS AN INPUT IN FORM (YEARS FROM 1900)(MONTH OF YEAR)
C      (DAY OF MONTH
C      WRITTEN AS 6501.12 FOR JAN. 12, 1965
C      T2 IS AN INPUT IN FORM (HOUR OF DAY)(MINUTE OF HOUR)
C      (SECOND OF MINUTE)
C      WRITTEN AS 1301.3032 FOR 1PM, 1 MINUTE, 30.32 SECOND
C      T3  IS WHOLE DAYS FROM 1950 OUTPUT
       CALL SETN(NIN,NOUT)
       T4=0.
       TA=INTF(T1)
       TA1=TA/100.+.01
       TY1= INTF(TA1)
       TY= TY1-60.
       IY=TY
       IF (IY) 15,15,16
 15    CONTINUE
       WRITE OUTPUT TAPE NOUT,17
 17    FORMAT (37H EXIT FROM TIMEC. DATE 1960 OR BEFORE)
       CALL EXIT
 16    CONTINUE
       ND=3652
       K=1
       DO 1 I=1, IY
       KK=K-I
       IF (KK) 2,2,3
     2 CONTINUE
       ND=ND + 366
       K=K+4
       GO TO 1
     3 CONTINUE
       ND=ND +365
     1 CONTINUE
       TY2=TA-TY1*100.+.1
       IY2=INTF(TY2)
       IF (IY2) 10,10,11
    11 CONTINUE
       IF (12-IY2) 10,12,12
    10 CONTINUE
       WRITE OUTPUT TAPE NOUT,13
 13    FORMAT(37H INPUT TIME IS WRONG. EXIT FROM TIMEC)
       CALL EXIT
    12 CONTINUE
       DO 4 I=1,IY2
       I=I
C            JAN FEB MAR APRIL MAY JUNE JULY AUG SEPT OCT NOV
       GO TO (4,6,7,6,8,6,8,6,6,8,6,8),I
     6 CONTINUE
       ND=ND + 31
```

```
      GO TO 4                                                           TIMC0480
    7 CONTINUE                                                          TIMC0490
      IF(KK-1)9,14,9                                                    TIMC0500
   14 CONTINUE                                                          TIMC0510
      ND=ND +29                                                         TIMC0520
      GO TO 4                                                           TIMC0530
    9 CONTINUE                                                          TIMC0540
      ND=ND +28                                                         TIMC0550
      GO TO 4                                                           TIMC0560
    8 CONTINUE                                                          TIMC0570
      ND=ND +30                                                         TIMC0580
    4 CONTINUE                                                          TIMC0590
      TB1=(T1-TA)*100.+.1                                               TIMC0600
      ND=ND +XINTF(TB1)                                                 TIMC0610
      ND=ND-1                                                           TIMC0620
      T3=ND                                                             TIMC0630
      TA=INTF(T2)                                                       TIMC0640
      T4=(T2-TA)/864.                                                   TIMC0650
      TB=TA/100.+.1                                                     TIMC0660
      TB1=INTF(TB)                                                      TIMC0670
      TB2=TA-TB1*100.+.1                                                TIMC0680
      TB3= INTF(TB2)                                                    TIMC0690
      T4=T4+TB1/24.+TB3/24./60.                                         TIMC0700
      RETURN                                                            TIMC0710
      END                                                               TIMC
```

Subroutine: TIMED

Purpose: To convert an input time from the format DAYS HOURS·MIN SEC to seconds. If the input is negative, it is assumed to be in seconds, and returned positive.

Calling Sequence:

CALL TIMED (T1, T2)

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	T1	1		-seconds	or (DAYS + 10^2 + HOURS + MIN
					x 10^{-2} + SEC x 10^{-4})
O	T2	1		seconds	

Common storages used or required: None

Subroutines required: None

Functions required: INTF

Approximate number of storages required:

```
*       LABEL                                                           TIMD
*       SYMBOL TABLE                                                    TIMD
        SUBROUTINE TIMED (T1,T2)                                        TIMD0000
        IF(T1)1,1,2                                                     TIMD0010
      1 CONTINUE                                                        TIMD0020
        T2=-T1                                                          TIMD0030
        RETURN                                                          TIMD0040
C       SUBROUTINE ACCEPTS T1 IN SEC IF THE SIGN IS INPUT NEGATIVE AND  TIMD0050
C       GIVES T2 IN + SEC IF T1 IS + AND ARRANGED AS (DAYS HOURS.MIN SEC) TIMD0060
C       IT CONVERTS OUTPUT T2 TO SEC                                    TIMD0070
      2 CONTINUE                                                        TIMD0080
        TDH=INTF(T1)                                                    TIMD0090
         TMS=T1-TDH                                                     TIMD0100
        TM1=TMS*100.+.01                                                TIMD0110
        TM=INTF(TM1)                                                    TIMD0120
        TS=(TMS*100.-TM)*100.+.01                                       TIMD0130
        TD=INTF(TDH/100.+.01)                                           TIMD0140
        TH=TDH-TD*100.+.01                                              TIMD0150
        TH=INTF (TH)                                                    TIMD0160
        TS=INTF(TS)                                                     TIMD0170
        T2=TD*86400. +TH*3600. +TM*60.+TS                               TIMD0180
        RETURN                                                          TIMD0190
        END                                                             TIMD
```

Subroutine: TRAC

Purpose: To obtain the vector from a body center to a tracking station and the unit up, east, and north vectors. The coordinate system is a rotating body fixed system. The body may be treated as an oblate spheroid.

Calling Sequence:

CALL TRAC (TO, AT, H, GHA, U, E, EN, RT, AC, SL, CL, ST, CT, A, B)

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
I	TO	1		RAD	Tracker longitude
I	AT	1		RAD	Tracker latitude
I	H	1		Km	Tracker Altitude
I	GHA	1		RAD	Greenwich hour angle
O	U	3	$\hat{U}$		Up unit vector
O	E	3	$\hat{E}$		East unit vector
O	EN	3	$\hat{N}$		North unit vector
O	RT	3			Radius vector to tracker

(Cont'd.)

Common storages used or required: None

Subroutines required: None

Functions required: SIN, COS, SQRT

Approximate number of storages required: 194 DEC

(Cont'd.)

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
O	AC	1			Constant used in program
O	SL	1			Sin LAT
O	CL	1			Cos LAT
O	ST	1			Sin LONG
O	CT	1			Cos LONG
I	A	1	R_E	Km	Body's equatorial radius
I	B	1	R_P	Km	Body's polar radius

Derivation of Vectors obtained in TRAC

A. Derivation of Unit U, E, and N

The coordinate system of concern is shown in Fig. 1 below.

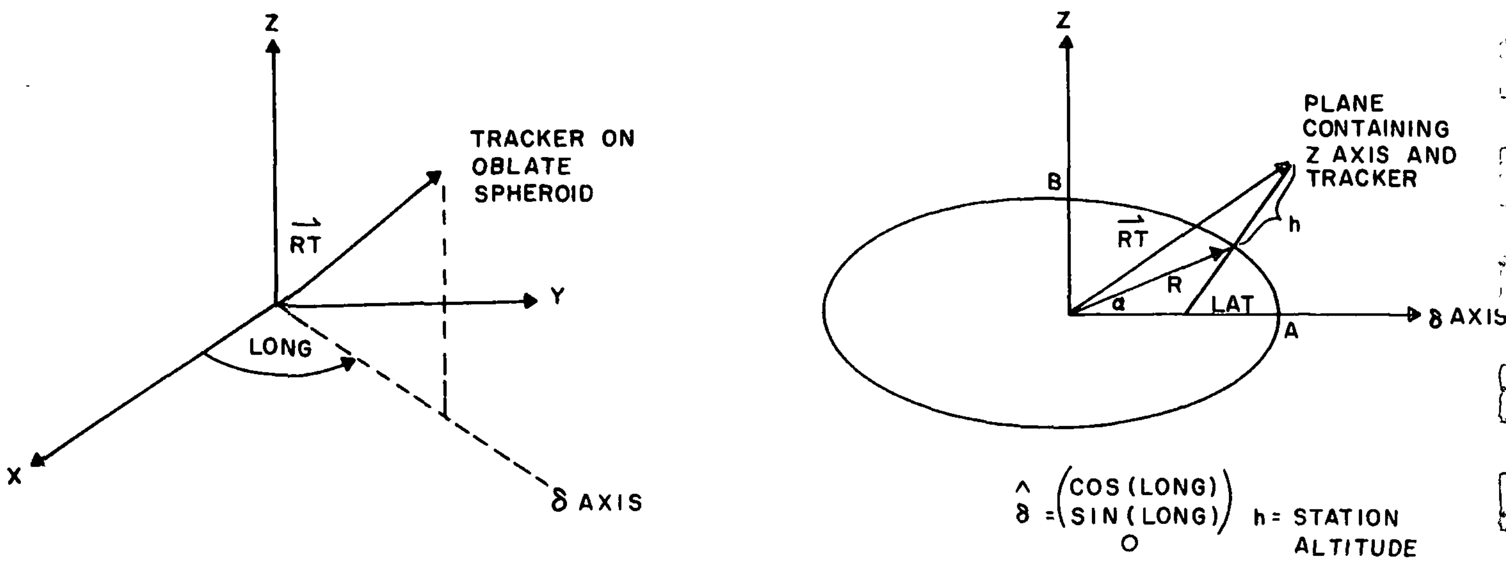


FIG I

The unit up vector, U, may be written as:

$$\hat{U} = \begin{pmatrix} \cos(LAT)\ \cos(LONG) \\ \cos(LAT)\ \sin(LONG) \\ \sin(LAT) \end{pmatrix}$$

The unit east vector, E, is:

$$\hat{E} = \frac{\hat{k} \times \hat{u}}{|\hat{k} \times \hat{u}|} = \begin{pmatrix} -\sin(LONG) \\ \cos(LONG) \\ 0 \end{pmatrix}$$

The unit north vector, N, is:

$$\hat{N} = \hat{U} \times \hat{E} = \begin{pmatrix} -\sin(LAT)\cos(LONG) \\ -\sin(LAT)\ \sin(LONG) \\ \cos(LAT) \end{pmatrix}$$

B. Derivation of Radius Vector, RT, to Tracker.

$$\overrightarrow{RT} = \vec{R} + h\,\hat{U}$$

$$\vec{R} = |R|\ \cos\alpha \begin{pmatrix} \cos(LONG) \\ \sin(LONG) \\ 0 \end{pmatrix} + |R|\ \sin\alpha \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

Equation of ellipse in Z-δ plane

$$\frac{Z^2}{B^2} + \frac{\delta^2}{A^2} = 1$$

Substituting for Z, δ

$$R^2\left(\frac{\sin^2\alpha}{B^2} + \frac{\cos^2\alpha}{A^2}\right) = 1$$

$$\frac{dZ}{d\delta} = -\frac{B^2}{A^2}\ \frac{\delta}{Z}$$

$$\text{Tan}(LAT) = \frac{A^2}{B^2}\ \frac{Z}{\delta}$$

$$\text{Tan}\ \alpha = \frac{Z}{\delta}$$

$$R = \frac{AB}{\left(A^2\sin^2\alpha + B^2\cos^2\alpha\right)^{1/2}} \qquad \therefore \ \mathrm{Tan}\,\alpha = \frac{B^2}{A^2}\,\mathrm{Tan(LAT)}$$

$$R\cos\alpha = \frac{A}{\left(\frac{A^2}{B^2}\,\mathrm{Tan}^2\alpha + 1\right)^{1/2}} \qquad R\sin\alpha = \frac{B}{\left(1 + \frac{B^2}{A^2}\cot^2\alpha\right)^{1/2}}$$

Substituting for Tan α yields

$$R\cos\alpha = \frac{A}{\left(\frac{B^2}{A^2}\,\mathrm{Tan}^2(\mathrm{LAT}) + 1\right)^{1/2}} \qquad R\sin\alpha = \frac{B}{\left(1 + \frac{A^2}{B^2}\cot^2(\mathrm{LAT})\right)^{1/2}}$$

$$RT(1) = \left\{\frac{A}{\left[\frac{B^2}{A^2}\sin^2(\mathrm{LAT}) + \cos^2(\mathrm{LAT})\right]^{1/2}} + h\right\} U(1)$$

$$RT(2) = \left\{\frac{A}{\left[\frac{B^2}{A^2}\sin^2(\mathrm{LAT}) + \cos^2(\mathrm{LAT})\right]^{1/2}} + h\right\} U(2)$$

$$RT(3) = \left\{\frac{B^2}{\left[B^2\sin^2(\mathrm{LAT}) + A^2\cos^2(\mathrm{LAT})\right]^{1/2}} + h\right\} U(3)$$

```
*       LABEL                                                            TRAC
*       SYMBOL TABLE                                                     TRAC
CEC2009                                                                  TRAC
        SUBROUTINE TRAC (TO,AT,H,GHA,U,E,EN,RT,AC,SL,CL,ST,CT,A,B)       TRAC000(
        DIMENSION RT(3),U(3),E(3),EN(3)                                  TRAC001(
C       TO AND LAT MUST BE INPUTED IN RAD,H IS ALTITUDE IN KM            TRAC002(
        SL=SINF( AT)                                                     TRAC003(
        CL=COSF( AT)                                                     TRAC004(
        TA=TO+GHA                                                        TRAC005(
C       GHA IS GREENWICH HOUR ANGLE,U IS UP,E IS EAST,                   TRAC006(
C       EN IS NORTH UNIT VECTORS                                         TRAC007(
        ST = SINF(TA)                                                    TRAC008(
        CT = COSF(TA)                                                    TRAC009(
        C = SQRTF(CL*CL+B*B*SL*SL/(A*A))                                 TRAC010(
        AC=A/C                                                           TRAC011(
        U(1) = CL*CT                                                     TRAC012(
        U(2) = CL*ST                                                     TRAC013(
        U(3) = SL                                                        TRAC014(
        RT(1) = (A/C+H)*U(1)                                             TRAC015(
        RT(2) = (A/C+H)*U(2)                                             TRAC016(
        RT(3) = (B*B/(A*C)+H)*U(3)                                       TRAC017(
      4 E(1) = -ST                                                       TRAC018(
        E(2) = CT                                                        TRAC019(
        E(3) = 0.                                                        TRAC020(
        EN(1) = -SL*CT                                                   TRAC021(
        EN(2) = -SL*ST                                                   TRAC022(
        EN(3) = CL                                                       TRAC023(
        RETURN                                                           TRAC024(
        END                                                              TRAC
```

Page Intentionally Left Blank

Subroutine: TRANSH

Purpose: To transform the partial matrix from equator and equinox of date to equator and equinox of 1950. The logic key is used to indicate if the last three partials are zero (LL = 1) or if all six partials have values (LL = 2).

Calling Sequence:

CALL TRANSH

Input and Output

I/O	Symbolic Name or Location	Program Dimensions	Mathematical Symbol	Data Dimensions or Units	Definition
IO	H	(1,6)	$\frac{\partial \text{ MEAS}}{\partial \text{ STATE}}$		Partials of measurements WRT*
					the state
I	LL	1			Logic key
					* With Respect To

Common storages used or required: T, S, C, IC

Subroutines required: None

Functions required: None

Approximate number of storages required:

```
*     LABEL                                                          TRAN
*     SYMBOL TABLE                                                   TRAN
      SUBROUTINE TRANSH(H,LL)                                        TRAN0000
C     SUBROUTINE TRANSFORMS H MATRIX TO EQUINOX 1950                 TRAN0010
      EQUIVALENCE(T,TDUM),(S,SDUM),(C,CDUM),(C(138),AN)              TRAN0020
      DIMENSION H(6),AN(3,3),DUM(1,6),T(1360),S(1000),C(1000),IC(1)  TRAN0030
      COMMON T,S,C,IC                                                TRAN0040
      LL=LL                                                          TRAN0050
      DO 1 I=1,3                                                     TRAN0060
      K=I+3                                                          TRAN0070
      DUM(I)=0.                                                      TRAN0080
      DUM(K)=0.                                                      TRAN0090
      DO 1 J=1,3                                                     TRAN0100
      DUM(I)=DUM(I)+AN(J,I)*H(J)                                     TRAN0110
      GO TO(1,2),LL                                                  TRAN0120
    2 CONTINUE                                                       TRAN0130
      L=J+3                                                          TRAN0140
      DUM(K)=DUM(K)+AN(J,I)*H(L)                                     TRAN0150
    1 CONTINUE                                                       TRAN0160
      DO 3 I=1,6                                                     TRAN0170
      H(I)=DUM(I)                                                    TRAN0180
    3 CONTINUE                                                       TRAN0190
      RETURN                                                         TRAN0200
      END                                                            TRAN
```

DISTRIBUTION LIST

Address	Number of Copies
National Aeronautics and Space Administration Goddard Space Flight Center Greenbelt, Maryland Attn: Dr. F. O. Vonbun	30
Philco Corporation WDL Division Palo Alto, California	70
	100